AFRL-IF-RS-TR-2004-278
Final Technical Report
November 2004

DEFENSIVE INFORMATION WARFARE TECHNOLOGY APPLICATIONS (DIWTA) AUTOMATED INTRUSION DETECTION ENVIRONMENT (AIDE) ADVANCED CONCEPT TECHNOLOGY DEMONSTRATION (ACTD)

Northrop Grumman Information Technology Inc.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2004-278 has been reviewed and is approved for publication

APPROVED: /s/

BRIAN T. SPINK
Project Engineer

FOR THE DIRECTOR: /s/

WARREN H. DEBANY, JR., Technical Advisor
Information Grid Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	NOVEMBER 2004	Final Aug 01 – Aug 03

4. TITLE AND SUBTITLE
DEFENSE INFORMATION WARFARE TECHNOLOGY APPLICATIONS (DIWTA) (AUTOMATED INTRUSION DETECTION ENVIRONMENT (AIDE) ADVANCED CONCEPT TECHNOLOGY DEMONSTRATION (ACTD))

5. FUNDING NUMBERS
C - F30602-99-D-0001/0001 & 0008
PE - 33140F
PR - AIDE, AIDE
TA - 01, JM
WU - 01, 08

6. AUTHOR(S)
Patricia Denno

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Northrop Grumman Information Technology Incorporated
7575 Colshire Drive
McLean Virginia 22102

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory/IFGB
525 Brooks Road
Rome New York 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2004-278

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Brian T. Spink/IFGB/(315) 330-7596/ Brian.Spink@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*
This report represents work done in Task 0001 and Task 0008.
This paper is organized into the three years represented. Introductory material provides an overview. Subsequent sections for each year begin by describing the development process which was based on incorporating the lessons learned from the previous year, the goals for the year, and including innovations from AFRL. The paper then describes for each year the deployment and demonstration planning processes. Demonstration performance is described and summary data from each year's demonstration follows. Finally user feedback and hot wash conclusions are then presented by year, leading to a discussion of suggested improvements for Year Three development. A final conclusions section completes the document.

14. SUBJECT TERMS
Intrusion Detection, Concept Technology, Integration Sites, System Engineering

15. NUMBER OF PAGES
63

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

TABLE OF CONTENTS

1 INTRODUCTION 1

1.1 THREE YEAR OVERVIEW 1
1.2 PHASE OVERVIEW 3
1.3 SUMMARY 3

2 YEAR ONE 4

2.1 YEAR ONE DEVELOPMENT 4
2.1.1 G-2 Interface Development 4
2.1.2 Oracle Database Development 5
2.1.3 Bridge Development 5
2.2 YEAR ONE DEPLOYMENT 5
2.2.1 Site Distribution 6
2.2.2 Sensor Distribution 6
2.2.3 Site Survey, Installation, and Testing 8
2.3 YEAR ONE DEMONSTRATION 9
2.3.1 Demonstration Planning 9
2.3.2 Demonstration Execution 9
2.3.3 Demonstration Data 10
2.4 YEAR ONE FEEDBACK 10
2.4.1 Combined Feedback from all sources 11
2.4.1.1 Indicators 11
2.4.1.2 Data Base 11
2.4.1.3 Browser 11
2.4.1.4 Network Mapping 12
2.4.1.5 Secure and G2-G2 Communications 12
2.5 RECOMMENDATIONS FOR YEAR TWO 13

3 YEAR TWO 14

3.1 YEAR TWO DEVELOPMENT 14
3.1.1 Developmental Goals for Year Two and Improvements Implemented 14
3.1.2 G-2 Interface Development 15
3.1.2.1 Correlation Rules 15
3.1.2.2 6510 Reporting 15
3.1.2.3 User Interface 15
3.1.3 Oracle Development 16
3.1.3.1 Normalization 16
3.1.3.2 Oracle Web Server 16
3.1.4 Bridge Development 17
3.1.4.1 Temporary Files 17
3.1.4.2 Efficiency 18
3.1.5 Other AIDE Development 18
3.1.5.1 Encryption 18
3.1.5.2 Training Program 18
3.1.5.3 Clock Synchronization and the Implementation of NTP 18
3.2 YEAR TWO DEPLOYMENT 19
3.2.1 Site Distribution 19
3.2.2 Sensor Distribution 20
3.2.3 Site Survey, Installation, and Testing 20
3.3 YEAR TWO DEMONSTRATION 21
3.3.1 Demonstration Planning 21
3.3.1.1 Test Planning Working Group 21
3.3.1.2 Measures of Effectiveness 21
3.3.2 Demonstration Execution 22

3.3.2.1 JC2WC Red Team Support ... 22
3.3.2.2 White Team Support ... 22
3.3.3 Year Two Demonstration Data ... 23
3.3.3.1 Demonstration Overview ... 23
3.3.3.2 Demonstration Data ... 23
3.3.3.3 Attack Generation ... 23
3.3.3.4 Data Capture: Sessions ... 24
3.3.3.5 Data Capture: AIDE Browser ... 25
3.3.3.6 Data Capture: White Cell Reporting ... 27
3.3.3.7 Data Capture: A Combinatorial Approach ... 28
3.3.3.8 GNOSC ... 30
3.3.3.9 Data Un-captured ... 30
3.3.4 Year Two Demonstration Technical Problems Encountered ... 30
3.3.4.1 NetRadar ... 30
3.3.4.2 Performance Problems ... 31
3.3.4.3 Configuration Problems ... 31
3.3.4.4 Database Inserts ... 32
3.4 YEAR TWO FEEDBACK ... 32
3.4.1 Pre-Demo (Training) Feedback ... 33
3.4.2 Demo (Site) Comments ... 33
3.4.2.1 Demo Prep and Execution ... 33
3.4.2.2 Three Best ... 33
3.4.2.3 Three Worst Things ... 34
3.4.3 Post-Demo (Developer) Comments ... 34
3.4.3.1 Performance ... 34
3.5 RECOMMENDATIONS FOR YEAR THREE ... 35
3.5.1 Development Team ... 35
3.5.1.1 Performance ... 35
3.5.1.2 Sensor Interface ... 35
3.5.1.3 User Interface ... 36
3.5.2 Hot Wash Suggestions ... 36

4 YEAR THREE ... 38

4.1 YEAR THREE DEVELOPMENT ... 38
4.1.1 Developmental Goals for Year Three ... 38
4.1.2 User Interface Development ... 39
4.1.2.1 New Java GUI ... 39
4.1.2.2 Web GUI ... 39
4.1.3 Oracle Development ... 40
4.1.3.1 Oracle Capabilities ... 40
4.1.3.2 New Tables and Fields ... 40
4.1.4 Bridge Development ... 40
4.1.4.1 Updated Bridges ... 40
4.1.4.2 New Bridges ... 40
4.1.4.3 Encryption ... 40
4.1.5 Correlation Development ... 41
4.1.5.1 Correlation System ... 41
4.1.5.2 Correlation User Interface ... 41
4.1.6 Other Aide Development ... 41
4.2 YEAR THREE DEPLOYMENT ... 41
4.2.1 Site Participation ... 41
4.2.2 Sensor Distribution ... 43
4.2.3 Site Survey, Installation, and Testing ... 43
4.3 YEAR THREE DEMONSTRATION ... 43
4.3.1 Demonstration Planning ... 43
4.3.2 Demonstration Execution ... 44
4.3.3 Demonstration Data ... 44
4.3.3.1 Demonstration Overview ... 45

4.3.3.2 Demonstration Data 45
4.3.3.3 Attack Generation 45
4.3.3.4 Data Capture: Sensors 47
4.3.3.5 Data Capture: AIDE Browser 47
4.3.3.6 Correlation 48
4.3.3.7 Data Forwarded 49
4.4 YEAR THREE FEEDBACK 50
4.4.1 Pre-Demo (Training) Feedback 50
4.4.2 Demo (Site) Feedback 50
4.4.3 Post-Demo (Developer) Feedback 51
4.5 RECOMMENDATIONS FOR FUTURE ENHANCEMENTS 52
4.5.1 Development Team 52
4.5.1.1 Easy Installation 53
4.5.1.2 Performance 53
4.5.1.3 Reliability 53
4.5.1.4 Operational Utilities 53
4.5.1.5 User Roles 53
4.5.2 DISA Compiled 54
5 CONCLUSION 57

LIST OF FIGURES

Figure 1-1: AIDE Timeline 2
Figure 2-1: Year One Participants 6
Figure 3-1: Year Two Participants 19
Figure 3-2: Measures of Effectiveness – High Level View 22
Figure 3-3: Year Two Demonstration Problem Categories Described by Week 32
Figure 4-1: Year Three Participants 42
Figure 4-2: Type of attacks captured by the AIDE database broken down by site 47
Figure 4-3: Number of Attacks on the AIDE browser broken down by Site 48
Figure 4-4: Site diagram of forwarded events 49

LIST OF TABLES

Table 2-1: Year One Sensor Distribution 7
Table 2-2: Year One Results 10
Table 3-1: Year Two Goals 14
Table 3-2: Sensor Distribution 20
Table 3-3: Number and Type of Network Attacks 24
Table 3-4: Type of attacks captured by the AIDE database broken down by site 25
Table 3-5: Number of Attacks on the AIDE browser broken down by Site 26
Table 3-6: Total Amount of Information available to the User during the Demonstration 28
Table 3-7: Matrix: Combination White Cell Reporting, Raw and Event Data Compared to Script 29
Table 3-7: Sensor Matrix Highlighting Additional Available Sensors at Year Two Sites 36
Table 3-8: Matrix of Hot Wash Suggestions and Design Improvements 37
Table 4-1: Year Three Goals 39
Table 4-2: Year Three Sensor Distribution 43
Table 4-3: Executed Attack Scenarios by Site showing sensor and AIDE detection 46
Table 4-4: Number of Sensor Events by Site 46
Table 4-5: Examples of reduction of events by correlation 48
Table 4-6: Year Three Developer Feedback 52
Table 4-7: DISA CCB list 56

1 Introduction

AIDE is the result of combined Defense Information Systems Agency (DISA) and Air Force Research Laboratories (AFRL) efforts to develop a computer network attack (CNA) warning capability which responds to the needs of local administrators, regional computer emergency response team (CERT) analysts, and analysts at the GNOSC. This warning capability is based on pushing, pulling, and fusing CNA information from multiple sources to create a local, regional, and global cyberspace view. To create this view, AIDE integrates distributed sensor data from legacy intrusion detection tools onto a single platform. From its first demonstration AIDE has successfully proved it was able to report intrusive behaviors to two levels, supplying a consolidated view at local and global levels.

1.1 Three Year Overview

This report encompasses the first three years of the ACTD where AIDE was developed from a concept to a fully operational prototype. The following two years will see the prototype developed into an operational system. Conceptually, this initial three year period breaks down into 3 one year segments or phases. In actuality, the time periods for each phase as separated by demonstrations varied from 6 to 18 months. The following figure shows the three phases and indicates some of the management changes over the entire duration

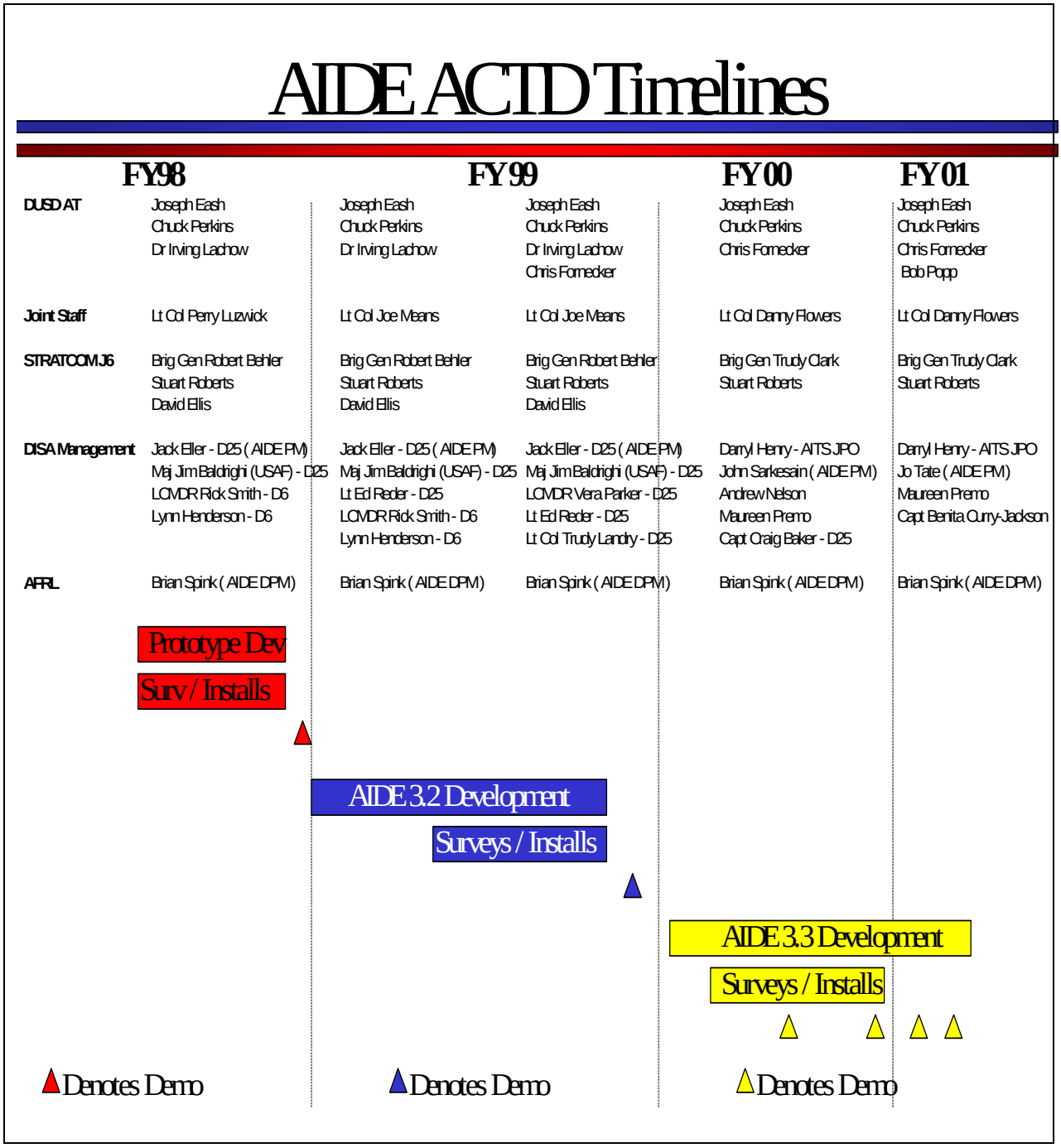


Figure 1-1: AIDE Timeline

1.2 Phase Overview

What follows in the next three major sections is a detailed description of the activities and outcomes of each of the three phases of the development activity.

For each phase by major subsection the major activities corresponding to each phase will be detailed. These major activities include:

Development – Those activities relating to the construction and development of the AIDE software and system. This section is divided into major components of the AIDE system such as the user interface, database, and sensor interface.

Deployment – Those activities relating to the deployment of the AIDE systems to specific test sites including lists of the sites participating, the sensors present at the sites, and the activities prior to, during, and after installation.

Demonstration – These activities that were part of the demonstration test for each phase of the AIDE development. Includes is a representation of the results of the demonstration.

Feedback – These activities include the collection and compilation of feedback from the developers, the end users, the results of post demonstration analysis, and the hot wash.

Recommendations – This section summarizes the suggestions for future changes and enhancements and serves as a conclusion based on the development activity that occurred, the deployment of the developed code, the execution of the demonstration and the feedback from those activities.

1.3 Summary

In the years to come AIDE will demonstrate the importance of CNA data correlation and data display to DOD's ability to orchestrate effective defensive responses. AIDE will also demonstrate the importance of information consistency across the CINCs, Services and Agencies, while providing an ability to "drill-down" to original supporting information. AIDE will empower security administrators to be aware of global CNA threats and to access locally relevant information from a single platform. In the years to come this tool, relying on continued sensor technology enhancements and the insertion of newer intrusion detection technologies, will enable administrators to quickly choose a course of action appropriate to the level of the attack. The ongoing development and demonstrations are essential to successfully developing a secure Defense Information Infrastructure (DII) in support of Joint Vision 2010.

2 Year One

The first year proved to be marked by aggressive goals for demonstration being balanced against difficult learning and development schedule. Although we refer to this period as year one, as noted in the introduction, the duration of time was approximately six months from the start of the contract to the completion of the first demonstration. Despite this, the first Information Assurance: Automated Intrusion Detection Environment (IA:AIDE) Advanced Concept Technology Demonstration (ACTD) was successfully conducted from14 to 25 September 1998. AIDE capabilities were evaluated at 7 sites in the continental United States, representing a cross section of DOD information networks worldwide. Feedback from these sites was positive. The first year goals of AIDE were to centrally display data from legacy sensors, and to pass this data both laterally between sites and up to the Global Network Operations Security Center (GNOSC). These goals were achieved. This achievement was the result of an initial development period exploring new software and ideas and integrating the two with existing software and systems to accomplish these goals.

2.1 Year One Development

Development in Year One was very much a learning process. Within a 6 month period it was expected that a test system would be designed, developed, and installed at sites throughout the country and expected to function. Using Gensym's G-2 environment to prototype the AIDE system and building on work already done on another program this goal would be met.

At the outset of year one and throughout the year, AIDE would be described as being composed of two parts: the primary interface and operational software written in G-2; and the "Bridges" to the sensors. Additionally data received from bridges and processed by G-2 was stored in an Oracle database. This would be the conceptual model that drove the development through Year One until the details of the problems inherent in the task we uncovered. The following two subsections detail development in the operational software written in G-2 and the "Bridges" to the sensors.

2.1.1 G-2 Interface Development

Using G-2's expert software, initially developed for process control in manufacturing, was a challenge. Our developers needed to understand its rules and limitations and apply them to network operations. Sensor data needed to be imported into G-2 using a process called bridging. Our developers learned the basics of bridging from AFRL's Extensible Program for Intrusion Control (EPIC). Rules needed to be applied to verify the data being bridged. Again EPIC personnel provided the baseline from which we developed the rules. Finally, data would be stored in the Oracle database.

2.1.2 Oracle Database Development

During this period the basics of the database schema as well as the G-2 interface to Oracle were developed. The primary function of the database was only a repository for historical data collected and processed by G-2. Basic tables were developed from the information that would be supplied from each of the sensors to be displayed via AIDE.

2.1.3 Bridge Development

The construction of bridges was based directly on sensor selection, which was based on a network survey of seven DOD installations. DISA tasked us to examine and bridge the data from sensors at these installations and to bridge data from the DTF (which was still in development and not initially deployed). We first acquired detailed knowledge of the individual sensors. Then, we built the bridges. Then we refined the G-2 browser interface to display the data. The September demonstration was the first large scale test of this G-2 configuration outside of the laboratory.

The September demonstration proved that intrusion detection and network management sensor data could be "bridged" into G-2 and the associated display would have value to systems security personnel. The demonstration highlighted two aspects of bridge development, which require improvement.

First, the reliability of the bridges and the sensors need to be improved. We found that several of the bridges and or sensors died during the demonstration. This usually meant killing and restarting the sensor and its associated bridge. Techniques for reestablishing the connection between bridges and sensors as well as restarting, both, required further study and would become goals for year two.

Second, the use of temporary files on remote sensors needed to be done very carefully and only when necessary. There was concern whether or not the remote sensor bridge would prevent the sensor from functioning. While running, some bridges created a temporary file on the sensor platform, and this file was overwritten when data was passed to AIDE. The demonstration has caused us to ask if AIDE were shutdown or if the connection with the bridge was broken, would the size of the temporary file continue to grow? In the worst-case scenario, the /var partition on the sensor could become full and cause the sensor to stop functioning. When temporary files are necessary, logic needed to be incorporated into the bridge so that when the connection with AIDE is broken, data was no longer written to the temporary file.

2.2 Year One Deployment

We surveyed networks of seven DOD installations. These installations served as a representative sampling of DOD networks. We had a unique opportunity to interact with site security and systems administrators at these locations. The AIDE program benefited immensely from technical and policy discussions with site personnel. Their feedback and cooperation proved invaluable to the continued success of this program.

2.2.1 Site Distribution

The initial site selection consisted of 7 sites. The reporting hierarchy consisted of three levels. The local level had AFRL reporting to AFIWC as well as STRATCOM and JFCOM reporting to Columbus R-CERT. The regional level included ASC as well as Columbus R-CERT and AFWIC all reporting to GNOSC. This is shown in figure 2-1.

We found the personnel at the individual sites very knowledgeable on the operation and security of their networks. We also found, however, that these sites had only a limited understanding of intrusion detection technology. Bounded by cost and service-specific standardization efforts, individual sites deployed only a limited number of intrusion detection tools. Most sites have limited access to intrusion detection data, relying for the most part on service CERTs and DISA's ASSIST for intrusion detection analysis. Attack correlation and warning were not done locally.

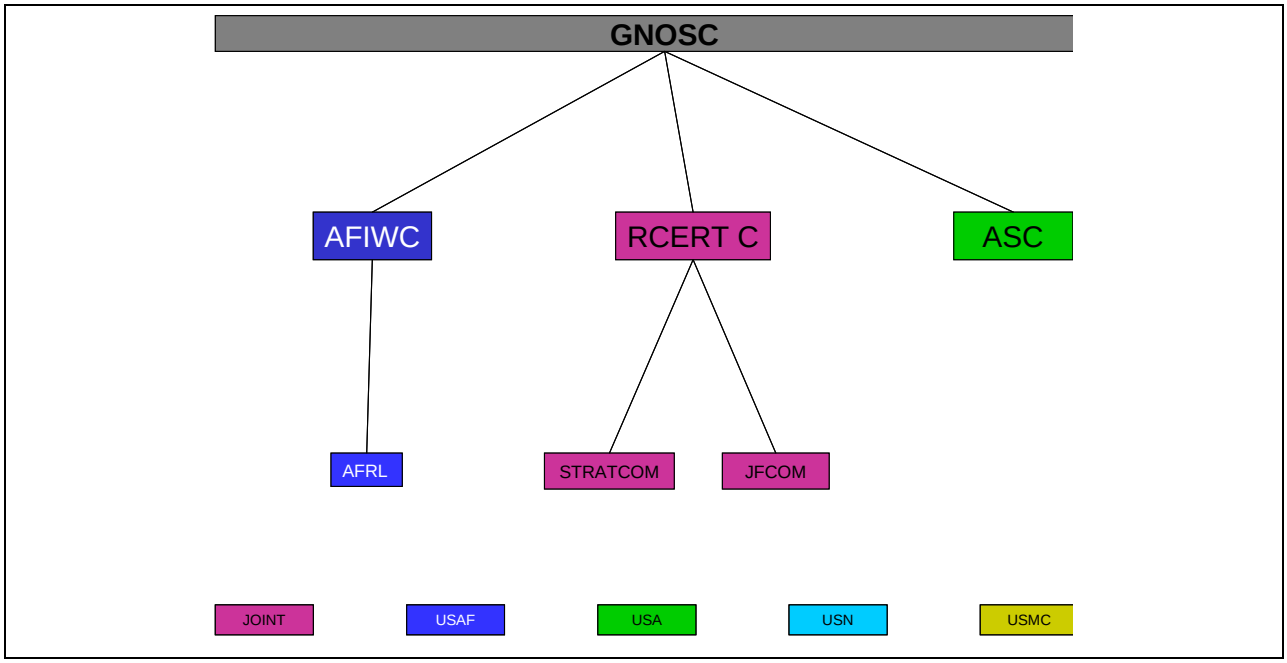


Figure 2-1: Year One Participants

2.2.2 Sensor Distribution

Prior to conducting the site surveys, DISA had assigned the ACTD a list of 33 sensors to integrate into AIDE. The list was made up of commonly known intrusion detection technologies, firewalls, network management tools, and virus checkers. The site surveys demonstrated that only a handful of items on the list were actually deployed. We had to make adjustments to ensure each site would be capable of reporting data to the GNOSC.

At four of the seven sites the Joint Intrusion Detection System (JIDS) was their only means of detecting anomalous network behavior. As DTF was to incorporate JIDS into its mix of sensors, we were precluded from integrating deploying a JIDS bridge beyond our laboratory. This left us with four sites with no sensors to feed AIDE. Consulting with EPIC developers we selected NetRadar as stopgap measure. NetRadar, an intrusion detection and intrusion reaction tool, is being developed under contract to AFRL and as such we were given ready access to the code. NetRadar had performed well at numerous EPIC demonstrations. We made NetRadar available to all seven sites.

Year One saw a distribution of sensors for AIDE found at the sites from the list of 33 targeted to be integrated into AIDE. Table 2-1 shows the sensor distribution for Year One by site.

Site \ Sensor	NetRadar	FPING	Strobe	NetRanger	Real Secure	HP OV	Raptor	ASIM 1.7	TCP Wrappers	Sunscreen
GNOSC	✔	✔	✔			✔				
AFIWC	✔	✔	✔					✔		
ROSC	✔	✔	✔							
ASC	✔	✔	✔		✔	✔			✔	
STRATCOM	✔	✔	✔				✔			
ACOM	✔	✔	✔							
AFRL	✔	✔	✔	✔		✔		✔	✔	✔

Table 2-1: Year One Sensor Distribution

Our ability to integrate firewall lag data was affected by a DOD firewall controversy. Based on command direction, some sites were relying on IP filtering in the absence of firewalls. We found some had disabled the firewall without substituting a similar capability. As most sites were in the process of either defining their new requirements or getting newer firewall configurations certified, we limited ourselves to bridging the SUNSCREEN firewall on-site.

We also found that three of the seven sites had no network management capability. We selected FPING and Strobe freeware products as stopgaps after consulting with AFRL. Both had been tested as part of the EPIC. FPING and Strobe were made available to all participants.

While we learned a great deal in our discussions with site personnel, it became apparent both during the installation of AIDE and during the demonstration that certain topics needed to be discussed in more depth. We need to take the time to work with local AIDE points of contact to

optimize the location of AIDE workstation within a site's network architecture. Network topology along with sensor type and location are critical factors to AIDE's success. The demonstration showed that we did not optimize the system's location, receiving only limited sensor data from a select number of sites. The system needs to be located where it can either see enough of the network to get a representative feed, or be directly connected to sensors that provide intrusion detection information. Future surveys and deployment planning will focus on proper positioning and connectivity to sensors, thereby ensuring AIDE provides relevant intrusion detection warning.

As part of the survey, we also need to get specific information on the volume of network traffic and the number of workstations on the subnet that AIDE is going to monitor. This information, coupled with the type of sensors used by a site, will influence the hardware requirements for a site's AIDE workstation.

In working with the sites, we need to clarify that in order to get data from remote sensors, a bridge needs to be physically installed on the workstation where the sensor resides. We understand the issues involved with running code on sensor workstations, and are working to come up with better solutions. Until that time, we are working to make the current solution as secure and reliable as possible.

2.2.3 Site Survey, Installation, and Testing

We deployed AIDE to seven sites in less than seven months. This accelerated schedule caused some problems. It was apparent during the installation of AIDE that the software baseline needed to be frozen several weeks prior to fielding. It was discovered that NetRadar was not generating audible alarms during zone changes. This was corrected at USACOM and USTRATCOM. By freezing the baseline earlier, we would have been able to conduct formal, rigorous testing prior to fielding and this error would have been caught. Additionally, installation verification procedures needed to be developed so that they can be exercised as part of the AIDE installation process. If these procedures are detailed, they can also serve as a training document. We have taken for action refining the installation process and to document the process.

As part of this process refinement, we need to reexamine the issue of remote AIDE management. To facilitate upgrading and adding capability to the system, we would like permission for AFRL to do remote configuration, installation, and monitoring. Using standard Unix capabilities and capabilities provided by G2, we are able to install, configure, and monitor a site's AIDE system. This capability would allow us to add capability to the system while limiting the inconveniences of sponsoring visitors to the site. Implementation will entail DISA GNOSC concurrence and a review of individual site policies.

The accelerated schedule made in-depth AIDE testing prior to deployment impossible. We need to develop a scripted set of computer network attacks. The purpose of these scripts would be to verify that AIDE system is properly configured to display sensor data. In addition, a site could use these scripts to conduct out of cycle demonstrations and to test site-specific configurations. During the demonstration we scripted a series of computer network attacks at the GNOSC and

launched on USACOM. These scripts verified that NetRadar was functioning properly. An added benefit of this controlled test was discovering that the AIDE browser had a 64K buffer limit. We learned that during a thorough port scan of a subnet, the AIDE browser requires additional space. In the near term we will be working with the GNOSC and service CERTS to develop these scripts.

We initially planned for three visits per site (site survey; installation; and demonstration). We found that with the exception of two sites, the other five required an additional visit to upgrade and maintain the software. Time and budgetary constraints will prevent us from visiting a new site more than three times within a given demonstration year. We will also need to limit visits to upgrade Year One sites. We will be working with the individual sites to develop a means to upgrade and test software remotely to preclude an extensive travel schedule.

2.3 Year One Demonstration

2.3.1 Demonstration Planning

AIDE was demonstrated with little prior coordination or establishment of clear objectives. Planning and communication are two keys to making the second AIDE demonstration even more successful. Briefing and coordination of demonstration details at all levels need to be provided well ahead of time, allowing participants enough time to work the details through their respective chain of commands. Roles and responsibilities of all parties need to be explicitly defined and coordinated with each participating site. A CONOPS discussing both the demonstration and the role of AIDE in network architecture needs to be written which incorporates many of the above details.

A detailed plan of the computer network attacks should be provided well ahead of time. The plan should include the location, the type of script run, and a general time frame. The participants need to have time to review and comment on the attack plan. A major difficulty with the first demonstration was getting information to provide deconfliction with the sites on real world versus demonstration related incidents. Providing the proper information will eliminate this problem as well as providing a training opportunity for the users.

2.3.2 Demonstration Execution

The first AIDE demonstration was successfully conducted from14 to 25 September 1998. AIDE capabilities were evaluated at 7 sites in the continental United States, representing a cross section of DOD information networks worldwide. Feedback from these sites was positive. The first year goals of AIDE were to centrally display data from legacy sensors, and to pass this data both laterally between sites and up to the Global Network Operations Security Center (GNOSC). These goals were achieved. The lessons learned from the demonstration were divided into two distinct groupings: management and technical implementation. Management lessons included the need for a more focused approach to conducting the demonstration and the creation of

measures of effectiveness. Technical lessons learned included creating more stable BRIDGE code (between the sensors and the AIDE system) and implementing stronger encryption for AIDE-to-AIDE communication.

2.3.3 Demonstration Data

Feedback from the individual sites was not standardized. To better pulse sites on their concerns we need to develop an accurate means of surveying them in a consistent and documented manner. Due to the discrepancies in the test plan, we did not have sufficient information from which to devise an operationally relevant survey. In future demonstrations our process will include early development of a test plan, developing operationally relevant test scripts, developing survey questions based on the two previous, and creating an easy to use, web based survey mechanism.

SITE	NUMBER OF PARTICIPATING SENSORS	TOTAL NUMBER OF RED TEAM ATTACKS	TOTAL NUMBER OF AIDE ALERT
AFRL	5	13	2
ACOM	3	19	17
ASC	4	24	15
AFIWC	4	27	0
ROSC-C	3	5	0
GNOSC	3	21	0
STRATCOM	1	27	0

Table 2-2: Year One Results

2.4 Year One Feedback

The September demonstration was AIDE's first field test. We deployed personnel to each of the seven sites to assist in AIDE 's operation, to record observations, and to solicit feedback from the individual sites. During the demonstration, many good ideas that would improve AIDE surfaced. The sections that follow briefly describe several of these ideas.

2.4.1 Combined Feedback from all sources

Note that during the Year one demonstration no distinction was made as to the source of these suggestions. Many are the result of combined effort on the part of the developers and site participants and occurred during the time spanning the initial installation to post demonstration data analysis.

2.4.1.1 Indicators

The AIDE demonstration helped solidify requirements to enhance system indicators. Improvements will include:

An indicator as to whether audible alarms are on or off;
An indicator when bridge connections to sensors are established;
An indicator when disk space is running low; and
An indicator when a G2 connection to GNOSC/ROSC is established.

2.4.1.2 Data Base

The choice of an Oracle database was a departure from the original EPIC and the DTF programs. The demonstration proved that pushing network intrusion detection data to the resident database was a desirable method of storing data for later analysis. Using the stored data proved more difficult. To avoid the need for advanced Oracle training we believe that a web-based front-end to the resident database is needed. This would facilitate analysts in retrieving session and alarm data. To ensure the longevity of this database we also need more robust, stand-alone backup scripts.

2.4.1.3 Browser

The AIDE browser displays general systems status information and alert information. During the demonstration we found that the AIDE Message Browser did not help apprentice system administrators determine whether emergency response was required. Skilled security administrators could understand what the browser display. A number of enhancements need to be made to the AIDE browser to make the data more useable for all skill levels. These include:

The manner by which sensor repetitions are displayed. Feedback from individual sites and AIDE personnel showed that it was difficult to understand when a new attack was beginning. Data from multiple sensors was rolled into a single entry to demonstrate first level data fusion. Users preferred that each sensor report be listed as a separate entry.

More complete data needs to be displayed in the browser. This may involve in routing sensor data first to either the database or the knowledge base prior to displaying it in the browser. By routing data to either the knowledge base or database, the volume of messages displayed in the browser may be reduced. The reduction would take place because data may be displayed only after knowledge base rules are met or database triggers have been activated. This “filtering” of data will also result in better descriptive

messages. For example, currently Net Radar indicates a port scan as a zone-change. Once the database has determined that a port scan is in progress, it could display a more accurate message.

Be more user friendly. The browser should facilitate automatic scrolling and easily deselecting highlighted messages. A double click should be all that is required to view an entry. We need to expand the use of colors in the browser to indicate the potential severity and type of sensor.

2.4.1.4 Network Mapping

The current method of mapping a site's network needs to be reworked. During the initial planning phase, DISA asserted that HP OpenView was deployed throughout DOD. An quick survey of potential sites showed that the Air Force was in the process of installing the software at its Network Operating Centers. The Air Force, however, would not allow the software to be used in a testing situation due to its operational mission. Of the seven sites, only one had an HP OpenView available for use. In future tests, for sites with DTF (which includes HP OpenView) we may be able to bridge HP OpenView data into IA:IADE. For sites without DTF, additional functionality will be required and we will have to improve the method used to display hosts on the screen.

2.4.1.5 Secure and G2-G2 Communications

We investigated a number of communications options during the initial development phase. Of the seven sites, three had no specific policy on how, or in what format, information on intrusions would be passed another organization. The remaining four required encrypted email be sent to their CERT. Initial investigations into purchasing virtual private network (VPN) technology were discontinued due to budgetary constraints.

We opted to secure all communications using the Simple Key management for Internet Protocols (SKIP). to provide secure point-to-point communications. IADE relies on SKIP to feed subsequent browser information to other AIDE boxes, both laterally and horizontally along DOD hierarchies for enhanced warnings and notification. SKIP secures the network at the IP packet level. Any networked application gains the benefits of encryption, without requiring modification. SKIP is unique in that an Internet host can send an encrypted packet to another host without requiring a prior message exchange to set up a secure channel. SKIP is particularly well suited to IP networks. All sites agreed to our using SKIP's 56-bit version. The SKIP met our cost goals, but was limited in capability. To improve the capability we will be using 128-bit version of SKIP in the next demonstration.

The initial communications schema included lateral communication between individual sites. While we designed the system to support AIDE-to-AIDE lateral communications, the sites requested DISA not insist on testing this capability. We successfully demonstrated the lateral communications capability between the Rome Research Site and the Air Force Information Warfare Center. We were not able to determine from this limited demonstration whether lateral

communications provides value-added information. To be able to understand the benefits of lateral information sharing, we should expand the number of participants in a future demonstration.

We succeeded this year to demonstrate that G2 could communicate and pass data hierarchically. Communications and data passing, however, required a high UNIX skill level. To increase AIDE's utility, G2-G2 communications need to be improved so that the connection between the individual sites and the GNOSC/ROSC can be automatically established. The functionality should include a user interface that allows the site to configure when and under what conditions the connection would be established. The reporting functionality should also include more capability for the user to report manually or automatically.

2.5 Recommendations for Year Two

According to the plan, by the end of the third demonstration the AIDE will reduce false positive reporting and to create a tactical warning capability. This first demonstration proves that we are well on the way to achieving that goal. The modest goals for the first demonstration of the AIDE ACTD were met. AIDE allowed local site analysts to receive, view, and analyze intrusion detection data. At four of the seven this was a significant increase in local capability. Each site detected testing activities (intrusions and map attempts) and was provided a timely warning. Feedback from the seven sites was positive and constructive. We believe there area number of opportunities to improve the system while providing a value added to the individual sites and to the global visualization endeavor. Cooperation with DISA and AFRL programs is essential to continued success and eventual fielding of AIDE technologies.

3 Year Two

The year two development began immediately following the year one demonstration in late September 1998 and lasted a period of approximately 12 months until the year two demonstration in August of 1999.

3.1 Year Two Development

At the outset of year two and throughout the year, AIDE would continue to be described as being composed of three parts: the primary interface and operational software written in G-2; the "Bridges" to the sensors, and the Oracle database. The immediately following section outlines overall goal as a function of the recommendations and lessons-learned from Year One. The remaining subsections detail development in each of the three primary parts as well as a section devoted to other components relating to AIDE.

3.1.1 Developmental Goals for Year Two and Improvements Implemented

The second AIDE demonstration built on the lessons learned in year one and expanded the focus to include a three-tiered reporting structure, improved visualization, and near-real time event correlation. This section describes the improvements made to the AIDE system. These improvements can be divided into three groups: 1) new functionality as part of the goals for Year Two; 2) improvements based on the lessons learned from Year One; and 3) innovations and enhanced functionality without specified requirement.

	Year Two Goal	Lesson Learned	Innovation
3 Tier Reporting	✔		
Normalization and Correlation	✔		
Oracle Web Server			✔
6510 Reporting			✔
Encrypt		✔	
Visualization		✔	✔
Bridge Development		✔	✔
Training		✔	
NTP		✔	

Table 3-1: Year Two Goals

3.1.2 G-2 Interface Development

3.1.2.1 Correlation Rules

As an enhancement to AIDE, implementing correlation was a requirement for Year Two development. The object was to implement rules across the sites in order to improve intrusion detection across the AIDE network. The first step was to normalize the sensor data. Sensor data was normalized within the database. Normalization was accomplished by comparing sensor data to a table known network activity. This activity was then distilled into a single reference. Normalization resulted in a standard terminology for all identifiable activity captured by the individual sensors. Once data was normalized rules were implemented within G-2 to capture events and compare them against other events within G-2. Should the comparison reveal a match, and then a correlated event was generated. A correlated event is described as the following:

- Duplicate events across sensors,
- Different events across sensors,
- Disparate sets of events across sensors, and
- Browser events across sites.

3.1.2.2 6510 Reporting

As an additional enhancement, AIDE is the first intrusion reporting capability within DOD to meet the intrusion reporting requirements outlined in Joint Staff's instruction " Defensive Information Operations Implementation" (CJCSI 6510). The developers instituted the capability to allow the user to input information relevant to the intrusion. The AIDE information report follows the format specified in Annex A to Appendix G to Enclosure D of Change 1 to *Defensive Information Operations Implementation,* CJCSI 6510.01B. Reports are automatically forwarded to elements higher in the hierarchy (to a regional or global).

While developing the 6510 reporting capability, the AIDE also assigned reporting priorities to events in the normalization table in accordance with 6510. This ensured that all priorities outlined in the instruction were matched with intrusions and attacks in the database. This was the first instance of automating 6510 reporting requirements within DOD.

3.1.2.3 User Interface

The AIDE development team drew on the user feedback from Year One to improve the display capabilities of the main AIDE browser. While the functionality of the browser (to display alerts in a compact format) did not change, the overall look and drill down capability did change. With Year Two users were able to:

- View correlated events,
- View the composition of the correlated event, and
- Closer scrutinize events in the Oracle Web Server.

AFRL/IFGB had been involved in a number of network visualization initiatives. Two initiatives, TASCVISION and SECURE SCOPE, were incorporated into the AIDE program and made

available to all users in the form of an executable CD. Both provided a 3-D visualization capability to support users in understanding and distinguishing between normal and abnormal events. Both rely on AIDE to collect the data. TASCVISION is designed to assist regional and global level users. Secure Scope is designed to support primarily users at the local level. It should be noted that both were independent external additions to the G-2 user interface

TASCVISION (TV) is a 3-D visualization product from Litton TASC. It provided a general, portable means for displaying 3-D views of data to enhance a user's ability to detect and react to patterns in available data. TV is a distributed information management and presentation system. It is a component-based framework implemented in Java™ which enables information–based collaboration. The primary goals are to defeat information overload and integrate information stovepipes. Its major characteristics are that it is a component-based framework, that it enables information-based collaboration and that it is implemented entirely in Java™. It was customized to work with AIDE for this demonstration. TV currently runs under Windows NT.

Secure Scope is an innovative 3-D data presentation interface that is built on a 3-tier, distributed architecture by Applied Visions, Inc. It provides flexible data analysis and exploration of intrusion and network operations data. It uses an intuitive, configurable, graphical "framework" for displaying diverse data sets. Secure Scope supports flexible "query" definitions to assist the user in creating a graphical display of events captured by sensors and collected by AIDE. To assist in analysis, Secure Scope presents associations among data items and is capable of displaying the correlation of hidden data properties.

3.1.3 Oracle Development

3.1.3.1 Normalization

As described previously for G-2 correlation, sensor data was normalized within the database. Normalization was accomplished by comparing sensor data to a table known network activity. This activity was then distilled into a single reference. Normalization resulted in a standard terminology for all identifiable activity captured by the individual sensors.

3.1.3.2 Oracle Web Server

The Oracle Web Server provided an enhanced functionality. The web-based front-end to the resident Oracle database was developed to assist users in retrieving session and alarm data and to avoid the need for advanced Oracle training to access database information. The AIDE web interface provides a front-end interface to the AIDE database. The front end was designed to give the user another way to display data collected by AIDE. The user has the ability to query data collected by various intrusion detection sensors. Unlike the AIDE/G2 interface, which displays data only for a very small time period, the Web browser approach allows the analyst to query all the data at any given time. The web site also contains a number forms that can be used to maintain the database tables used by the AIDE system. The user has the ability to add, update and delete records from these tables. The user is able to query by:

- Destination IP,
- Destination port,
- Sensor name,
- Sensor session ID,
- Signature

Additionally the user is able to edit or delete data for certain tables, i.e.:

- Normalization table,
- Bad IP list,
- Registered domains/Registered IP masks, and
- The site table

An additional functionality of the web server built into the AIDE web server allows remote users at a regional or global site to drill down to the local site's database. All communications are encrypted using 128-bit encryption.

3.1.4 Bridge Development

The new Tap/Bridge format required rebuilding the Year One Bridges. New Bridges were developed for the new and additional. The new Bridges were: Sidewinder 4.1, Raptor 6.0, ASIM 2.0, JIDS 2.1, CISCO Router, CISCO PIX, NetRadar, RealSecure, Gauntlet, DTF, and TCP Wrappers. *Please note that while not all sensors at a given location were made available to the AIDE program, the team did build Bridges to accommodate all sensors at all the Year Two sites.*

3.1.4.1 Temporary Files

The Year One Hot Wash identified two aspects of Bridge development, which required improvement: BRIDGE reliability and not using temporary files. The AIDE Development Team made the following improvements to address this requirement. The team:

- Moved the core code of the BRIDGE off of the sensor system so that most data processing would be handled at the AIDE system. Testing showed that this approach improved performance and reliability.
- Created a heartbeat feature displayed on the central G-2 browser so users could monitor the status of sensor communication.
- The use of temporary files was eliminated to avoid data overload if a sensor should fail.
- Went to a TAP/BRIDGE format. The TAP resided on the sensor system and simply captured the real-time data, tagged it with the sensor name, and sent it to the BRIDGE via a TCP connection. The BRIDGE resided on the AIDE system and would read in the event data from the TAP, validate, parse, rebuild, and send the event to the knowledge base.
- Added test interfaces so that each interface would have a troubleshooting option.

3.1.4.2 Efficiency

Year Two innovations related to BRIDGE development centered on improving the efficiency of the data transmitted into the G-2 engine. The goal was to increase performance of data transmission and enhance data usability. The team:

- Changed to a connection-based server using Berkeley Sockets for the TAP to BRIDGE communication. These connections were found to be very fast.
- Built in a standard data-set feature to utilize for all event types.
- Created code to handle revolving data files without having to restart the interface.
- Added the capability for each TAP to be executed via command line options or default configuration settings.
-

3.1.5 Other AIDE Development

3.1.5.1 Encryption

As in Year One, AIDE used the Simple Key management for Internet Protocols (SKIP) to provide secure point-to-point communications. Secure communications were established hierarchically and laterally. Based on user feedback from the Year One participants, the team upgraded SKIP to the 128-bit version of SKIP to improve the security posture of the system. Using SKIP a user at regional and global levels could access local databases remotely, enabling the user to analyze local data securely.

3.1.5.2 Training Program

During Year One the AIDE team provided each site with a comprehensive operations manual during the installation of the system. This manual, while providing detailed information for the operators, was inadequate to meet site commanders' needs. Based on the lessons learned, the AIDE team devised a comprehensive training program to ensure users had hands-on experience with AIDE and understood their roles during the demonstration. A total of 41 users were trained, during 2 ½ day training sessions at AFRL. Each user received hands-on training on an AIDE system configured to reflect the capability at his/her site. In addition, all users were given both an AIDE manual and a training guide to prepare them for the demonstration.

3.1.5.3 Clock Synchronization and the Implementation of NTP

An important lesson learned in Year One was that different intrusion detection and barrier technologies did not have a requirement to synchronize times both within a particular site and across sites. Deconflicting events within the database became difficult as a variety of sensors produced a variety of timestamps. Year Two used the Network Timing Protocol (NTP) to synchronize AIDE event logging across all sites. The team enabled the individual AIDE boxes to communicate with AFRL's NTP timeserver. This facilitated correlation of events across multiple sites at RCCs and GNOSC, and significantly improved the time required to analyze traffic during the post-demonstration analysis.

3.2 Year Two Deployment

This section describes the deployment of the AIDE system prior to the demonstration and some of the information learned during the deployment.

3.2.1 Site Distribution

Year Three's site selection consisted of 12 sites. Of these, 6 were from year one's initial selection of sites with 6 new sites. As in year one, the reporting hierarchy consisted of three levels. The local level had AFRL, Offut, SPACECOM, and ACC reporting to AFIWC as well as STRATCOM and JFCOM reporting to Scott R-CERT. Additionally at the local level ASC was reporting to LIWA. The regional level included Scott R-CERT, AFWIC, FIWC, and LIWA all reporting to GNOSC. This is shown in figure 2.

It should be noted that the number of sites grew from seven to twelve (refer to Figure 3-1). As in Year One, we found the personnel at the individual sites very knowledgeable on the operation and security of their networks. We also found, however, that these sites had only a limited understanding of intrusion detection technology. Bounded by cost and service-specific standardization efforts, individual sites deployed only a limited number of intrusion detection tools. Most sites have limited access to intrusion detection data, relying for the most part on service CERTs and the DOD CERT for intrusion detection analysis. Attack correlation and warning were not done locally.

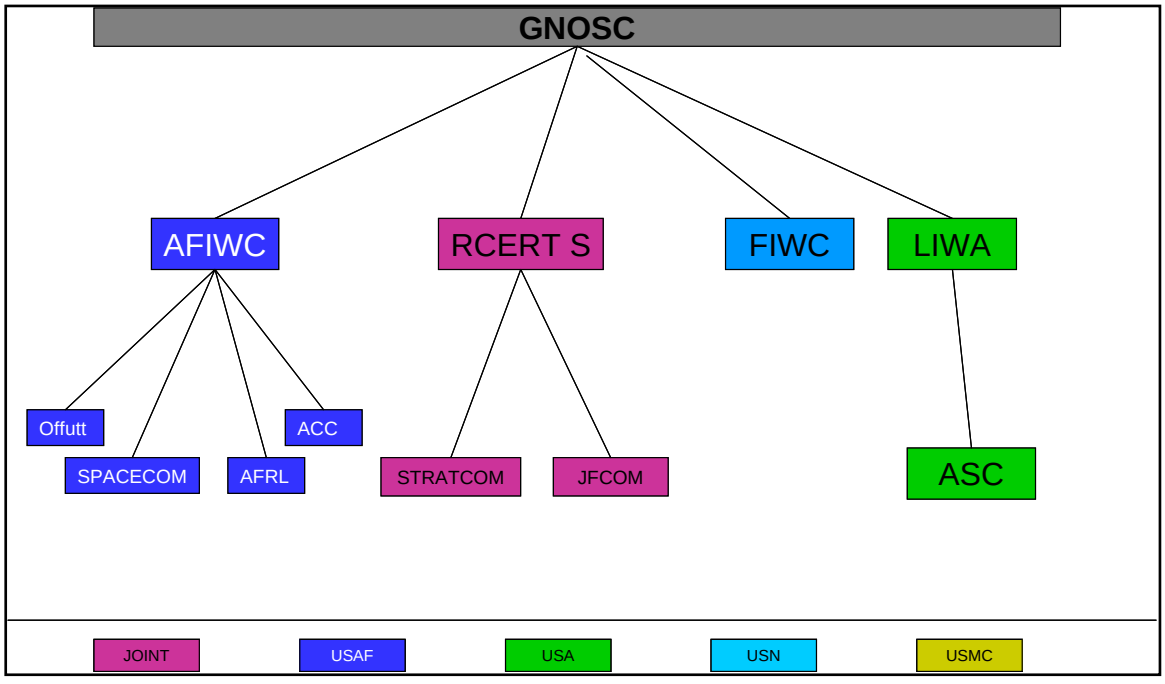


Figure 3-1: Year Two Participants

3.2.2 Sensor Distribution

Year Two brought an additional 5 sensors into AIDE and left only 4 of the initial 10. Table 3-2 shows the sensor distribution for Year Two by site. The new sensors are shaded as well as the new sites.

Site \ Sensor	NetRadar	JIDS	NetRanger	Real Secure	DTF	Raptor	Sidewinder	TCP Wrappers	Sunscreen
GNOSC	✔	✔						✔	
LIWA	✔	✔	✔	✔	✔		✔	✔	
SCOTT RCERT	✔			✔	✔			✔	
AFIWC	✔							✔	
FIWC	✔		✔	✔	✔			✔	
ASC	✔				✔			✔	
STRATCOM	✔	✔			✔	✔		✔	
ACOM	✔				✔			✔	
Offut	✔				✔		✔	✔	
ACC	✔			✔	✔		✔	✔	
AFRL	✔			✔	✔			✔	✔
SPACECOM	✔				✔			✔	

Table 3-2: Sensor Distribution

3.2.3 Site Survey, Installation, and Testing

The new participants were agreed upon by the end of January 1999. The sites were selected to include the service components to the newly established Joint Task Force for Computer Network Defense (JTF-CND) (LIWA, AFIWC, and FIWC), to include the lead CINC for Information Operations (SPACECOM), to include a service-specific enterprise management program (ACC and Offutt) and to normalize CINC intrusion incident reporting along the newly DISA Regional CERT concept (SCOTT R-CERT). Once the new sites had been agreed upon the deployment schedule ran as follows:

- 30 April 1999 - all new sites were surveyed and briefed on the AIDE program.
- 5 July 1999 - all new sites had received an operational AIDE system.
- 12 July 1999 - all upgrades to Year One participant's systems were completed.
- 7 August 1999 - all demonstration upgrades were completed.

3.3 Year Two Demonstration

The Year Two demonstration was conducted from 16 to 27 August 1999. All twelve sites participated. The Aide Team provided a technical support person at each site. DISA provided a Command White Cell to direct the demonstration. The Joint Command and Control Warfare Center (JC2WC) acted as the Red Team. This section describes the planning process and introduces the criteria used to measure the demonstration. This section concludes with a brief discussion on how and who conducted the demonstration.

3.3.1 Demonstration Planning

Lessons learned from Year One's demonstration highlighted the fact that future demonstrations needed considerably more management and planning prior to execution. Also emphasized was the need for more site involvement in planning the demonstration. DISA and AFRL facilitated site involvement by co-sponsoring 5 meetings to plan the demonstration, the Red Team requirements, and to associate the demonstration results to definitive measures of effectiveness.

3.3.1.1 Test Planning Working Group

The Test Planning Working Group (TPWG) comprised DISA, AFRL, STRATCOM, JC2WC, and the twelve sites. The strategy to planning the demonstration was to understand the Year Two development requirements and to test whether the finished prototype met Year Two goals. The TPWG developed the measures of effectiveness to quantify the success of the demonstration. The TPWG enlisted the support of the JC2WC to provide expertise at generating attacks. JC2WC created a series of scripted events, which comprise a subset of popular hacks and electronic reconnaissance tools. These events were mapped to the measures of effectiveness. The TPWG coordinated the scripts with the individual sites. The JC2WC traveled to Rome to test the scripts and ensure that AIDE was capable of capturing data from these scripts.

3.3.1.2 Measures of Effectiveness

The TPWG developed the measures of effectiveness to gauge AIDE's performance during the Year Two Demonstration. There were four general categories under which AIDE was evaluated: Integration and Interface, Correlation, Automated Warning, and Other. Figure 3-2 shows the breakout of the categories. For a detailed description of the measures of effectiveness refer to the Final Report for Year Two.

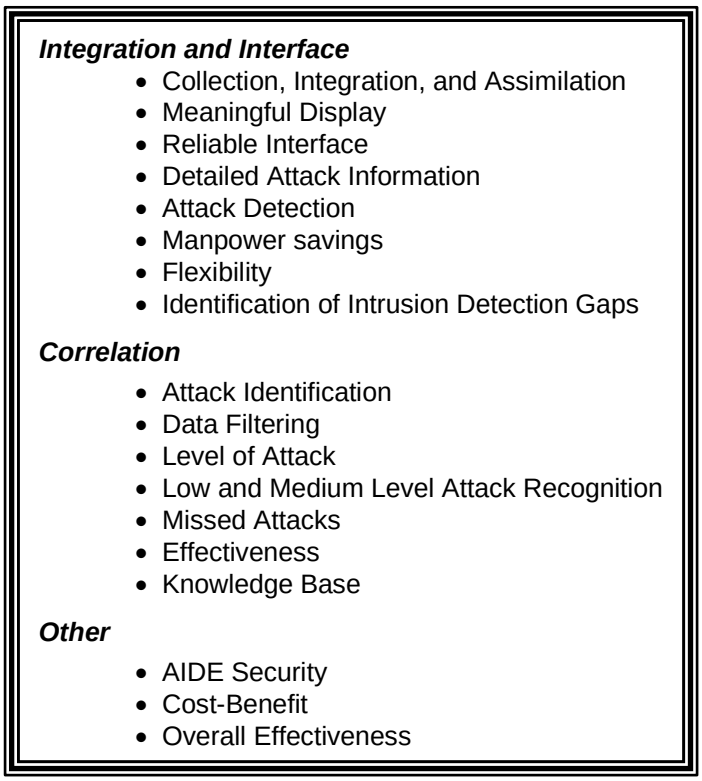


Figure 3-2: Measures of Effectiveness – High Level View

3.3.2 Demonstration Execution

There were three major players during the demonstration: the users at the individual sites, the Red Team, and the White Team. With the exception of the Air Force Information Warfare Center, all users were manning AIDE throughout the demonstration. The demonstration began at 1000 EDT on 16 August with a conference call between the Command White Cell and the White Cell members at all the sites. The first day concluded with a conference call between these players at 1600 EDT. Morning and afternoon conference calls continued throughout the demonstration to trouble shoot problems and to announce changes to the demonstration plans.

3.3.2.1 JC2WC Red Team Support

The JC2WC provided Red Team expertise and support. The Red Team generated scripted events against all AIDE sites during the specified times within the two week demonstration period. The Red Team "attacked" targets pre-selected by the individual sites. The target's complete name, domain name, and IP addresses were supplied to the Red Team prior to the demonstration.

3.3.2.2 White Team Support

The White Team functioned as a liaison between the Red Team and users. Each site supplied a White Team member. To support the demonstration and to manage attacks, DISA/D25 supplied a Command White Cell co-located with the Red Team. Prior to each day's attack, White Cell

member received the event scripts. This script enabled the White Team to validate Red Team activity and to deconflict test activity from real world operations.

3.3.3 Year Two Demonstration Data

This section contains the technical data extracted from the site databases after the demonstration. It discusses the successes the challenges experienced during the demonstration.

3.3.3.1 Demonstration Overview

AIDE users were able to access and use information relating to 15% of the events generated during the demonstration. This compares favorably with the GAO's 1996 review of intrusion detection systems within the federal government that claimed intrusion detection systems captured on average only 5 percent of attacks. This compares favorably to DARPA's findings of 20% in controlled laboratory tests. AIDE met it Year Two goals. AIDE was able to display attack data in a three-tiered hierarchy. Intrusion data was normalized and intrusion events were correlated at the local level.

While AIDE was able to exceed traditional detection rates for single sensors, it experienced performance problems and data management problems, which will require attention for Year Three. Data display continues to require refinement. Correlation was sluggish and the descriptions relating to the correlated event were difficult for the users to understand.

3.3.3.2 Demonstration Data

The first three days of the demonstration were primarily devoted to repairing technical problems, communications problems, and working out the event generation process. By Day 5 only two sites were experiencing persistent technical problems. Analyzing the demonstration data has been difficult in that data sets from individual site databases are incomplete. Problems with NetRadar and G-2 performance resulted in gaps in attack data at every site. Inconsistent reporting from White Cell members made filling in these gaps difficult. The following charts present the demonstration data for the entire 10-day demonstration. The data is presented from three points of view, the number of attacks by type, the number of attacks captured within the database, and the number and types of attacks displayed to the AIDE user. This data is fused to produce a comprehensive view of information available to the user during the demonstration.

This data does not contain the GNOSC information, which due to its role, was not subject to Red Team attacks. GNOSC data will be discussed separately later in the chapter.

3.3.3.3 Attack Generation

Table 3-3 shows the number and type of attacks that were generated against networks monitored by AIDE-connected sensors. This number does not include the insider, scripted attacks planned by the JC2WC. During post demonstration discussions it was discovered that a number of sites did not run the scripts. Few of the sites that did perform the insider attacks documented which

attacks they ran and when they were run. At only 1 of the 11 sites was DTF configured to report source and/or target IP. Unable to account for these attacks, the AIDE team decided not to count them either as attacks or as events captured.

Attack	Generated Total
NBTSTAT	52
IP SPOOFING	49
REMOTE EXECUTION	96
TELNET TO HTTP	77
CHECK X services	106
TELNET TO SMTP	50
TELNET TO GOPHER	77
SHOWMOUNT requests	127
FTP requests	195
PINGS	115
TRACEROUTES	83
PORTSCANS	80
NETBUS CHECK	151
CYBERCOP SCANS	98
SATAN SCANS	34
IDENT DAEMON CHECK	36
SNMP CHECK	96
SLAMMER	7
DOMAIN DUMP	105
PASSWORD GRINDING	41
TELNET PORT 23	71
RED BUTTON	42
FIREWALK	4
SAINT SCAN	1
OTHER	2
GRAND TOTAL	1795

Table 3-3: Number and Type of Network Attacks

Thresholds and ICMP traffic: Sensors associated with a given site are managed according to local policy. This means that reporting thresholds for similar attacks vary from site to site. PINGS, PINGS Sweeps, TRACEROUTE, and FTP requests were the most popular attacks that were filtered to the point of not being reported by sensors at particular sites. Most sites also enacted policies that blocked ICMP traffic at their firewalls.

3.3.3.4 Data Capture: Sessions

Table 3-4 depicts all the attack information collected in the AIDE database that could be mapped to a Red Team attack. This data was saved in the Oracle sessions' table. Sessions data relates to

all data captured using a network intrusion detection system, NetRadar. Unfortunately in both at STRATCOM, Scott R-CERT and Offutt the intrusion detection system was behind a firewall and could not collect supporting data. In the case of AFRL and FIWC a router blocked most connections to NetRadar, and severely limited the amount of session data collected.

	ACC	AFRL	ASC	FIWC	LIWA	Offutt	Scott	SPACECOM	STRATCOM	ACOM	AFIWC	Total
ANON FTP	3		4					2		4	1	14
Brute Force Pswrd Guessing	5	2								4		11
Cybercop	13	4	1					9		9	2	37
IDENTTCPSCAN			2					3			2	7
Maptool			10					13			9	32
NBTSTAT	1											1
Nslookup	1		1									2
Satan	9	2	1									12
Telnet	4		2					5		2	1	14
Telnet (21)	10	3						1		5		19
Telnet (25)	2									6		8
Telnet (70)	10	3								5		18
Telnet (161)	0									10	2	12
Telnet (6000)	9	2										11
Telnet (12345)	0									3	2	5
Telnet (12346)	0									2	2	4
Telnet (12345/12346)	0									7		7
Telnet (31337)	9							1		12	2	24
Xscan	1		6					2				9
Total	79	16	26	0	0	0	0	36	0	69	23	247

Table 3-4: Type of attacks captured by the AIDE database broken down by site

3.3.3.5 Data Capture: AIDE Browser

Table 3-5 depicts event data captured by AIDE and displayed on the AIDE browser. Discerning which data was generated in direct relation to a Red Team attack proved difficult in a number of cases. Problems with incomplete databases and non-descriptive sensor output made the post demonstration analysis difficult. Sidewinder firewall data only reported either “ACL Insert” or “ Netprobe”, to correlate this data to an actual attack required reviewing the timestamps and the targets. The Raptor firewall at STRATCOM only reported the time and source of an attack. Other sensors, while more descriptive, also provided incomplete data sets. It was hoped that in those cases where sensors provided incomplete data, other sensor data would help by providing additional insight into the attack. Event data is NetRadar attack data, AIDE correlated data, and

other sensor data that has been analyzed to match a pattern (in the case of NetRadar, Real Secure, and JIDS), or has been pre-analyzed by the sensor (in the case of firewalls) to be considered an attack. This data is saved in the Oracle database as "events". The figure shows that only a limited number of correlated events were saved into the database. In comparison to the chart above (figure 4.2.2), approximately half of the traffic was classified as an event. Only a few events generated from lateral or subordinate sites were actually classified as events among the regional sites.

Please note: One cannot compare the number of events displayed versus the number of sessions captured because local policies and reporting thresholds vary by sensor and vary between sites. Not all attacks generated by the Red Team warranted a display on the AIDE browser. The AIDE development team and the JC2WC did not coordinate which attacks would be viewed as an alert and at which level the alert would be displayed. This resulted in JC2WC scripting and executing attacks that, although captured by the database in the sessions' table did not display on the browser. The planning group accepted all the attacks generated by the JC2WC as activity normally experienced by a network systems administrator. The reason these attacks did not display is that reporting thresholds established both by vendors and local policies do not consider the specific type of attack actionable. These attacks included TELNETS; FTPs; scans and PING sweeps. Local sites agreed that it would have been undesirable to consider these groups of attacks with any greater seriousness than were handled by the AIDE system. Users saw value in the fact that such activity could be monitored within G-2 and should locally determined thresholds be met that the activity would have been reported as a correlated event.

	Events Captured	Correlated Events Captured	Lateral or Subordinate Events	Total Events
ACC	18	0	7	25
ACOM	2	1	3	6
AFIWC	6	0	0	6
AFRL	0	0	0	0
ASC	4	0	0	4
FIWC	1	1	0	1
GNOSC	0	0	42	42
LIWA	7	0	4	11
Offutt	39	0	0	39
Scott	0	1	9	10
SPACECOM	15	0	0	15
STRATCOM	7	0	0	7
Totals	99	3	64	166

Table 3-5: Number of Attacks on the AIDE browser broken down by Site

3.3.3.6 Data Capture: White Cell Reporting

Table 3-6 is the total amount of information the user viewed during the demonstration. This data was extracted from the daily White Cell reporting from all 11 reporting sites. The total of 277 is significant in that it represents what information the users actually accessed and analyzed to make decisions as to whether to alert network security administrators and whether to send a 6510 report to higher headquarters. There is a slight problem with this data in that some sites were more diligent than others in reporting the attacks viewed. The AIDE team made a conscious decision to display only priority one raw alerts and all alerts resulting from correlation. The decision was based on a reluctance to overwhelm the user with attack information that did not require immediate action. Initially those sites equipped only with NetRadar were seeing very few alerts. Many sites were not aware that they could view raw data (sessions) in the Oracle Web browser until 3 days into the demonstration. This resulted in not all attacks were reported. The thresholds were adjusted during the fourth and fifth days of the demonstration so that all raw alerts could be viewed. At sites with multiple sensors reporting high volumes of traffic (i.e. firewalls and Real Secure) this option was not exercised.

	ACC	AFRL	ASC	FIWC	LIWA	OFFUTT	Scott	SPACE COM	STRAT COM	ACOM	AFIWC	TOTAL
ANON FTP	3		2	4	12			6	1		2	30
Brute Force PSWRD Guessing	4	2			1							7
Cybercop	12	5		5	6			14			7	49
IDENTTCPSCAN		1	2	3	2			2				10
Maptool				4								4
NBTSTAT	1				1							2
Nslookup	1		1		2							4
Satan	8		2		2			1				13
Telnet	4		2						1		2	9
Telnet (21)	9				2			2				13
Telnet (25)	2		4		1			3				10
Telnet (70)	8											8
Telnet (161)												0
Telnet (6000)	9							2	3			14
Telnet (12345)												0
Telnet 12346)												0
Telnet (12345/12346)												0
Telnet (31337)	6								1			7
Xscan			7									7
Other		1										1
Traceroute		2		1								3
Sun RPC		1	5		1			2	1			10
Port 69					1							1
Http Port 80					2							2
Showmount			5		2							7
Ping and Ping Sweep					6				2			8
Rlogin											1	1
Unidentified Sidewinder				11	14	30						55
Total	69	12	30	28	55	30	0	32	9	0	12	275

Table 3-6: Total Amount of Information available to the User during the Demonstration

3.3.3.7 Data Capture: A Combinatorial Approach

This is *"a what if"* section: if the database had captured all of the attacks (either in sessions or events tables) we believe the matrix would have looked like table 3-7. During this post demonstration analysis the Team discovered a discrepancy between the number of attacks reported as having been seen (both in the browser and the web server) and the number of events and sessions actually available in the site's database. The matrix in Table 3-7 is an attempt to understand what data was lost in the database, but not lost to the user. This matrix comprises all session and event data in the database correlated against all the attacks reported as having been

	ACC	AFRL	ASC	FIWC	LIWA	OFFUTT	Scott	SPACE COM	STRAT COM	ACOM	AFIWC	TOTAL
ANON FTP	3		4	4	12			6	1	4	2	36
Brute Force PSWRD Guessing	5	2			1					4		12
Cybercop	13	5	1	5	6			14			7	51
IDENTTCPSCAN		1	2	3	2			3		9	2	22
Maptool			10	4				13			9	36
NBTSTAT	1				1							2
Nslookup	1		1		2							4
Satan	9	2	2		2			1				16
Telnet	4		2					5	1	2	2	16
Telnet (21)	10	3			2			2		5		22
Telnet (25)	2		4		1			3		6		16
Telnet (70)	10	3								5		18
Telnet (161)										10	2	12
Telnet (6000)	9	2						2	3			16
Telnet (12345)										3	2	5
Telnet 12346)										2	2	5
Telnet (12345/12346)										7		9
Telnet (31337)	9							1	1	12	2	25
Xscan	1		7					2				10
Other		1										1
Traceroute		2		1								3
Sun RPC		1	5		1			2	1			10
Port 69					1							1
Http Port 80					2							2
Showmount			5		2							7
Ping & Ping Sweep					6				2			8
Rlogin											1	1
Sidewinder Rep				11	14	30						55
Total	79	22	43	28	55	30	0	54	9	0	31	421

Table 3-7: Matrix: Combination White Cell Reporting, Raw and Event Data Compared to Script

seen by the White Cell or operators at the individual sites and compared to the attack script. This highlights where the user had the opportunity to make the most out of the data available and which sites and configurations need the most attention in preparation for Year Three's demonstration. The initial conclusions require us to look closely at the implementation of correlation, data capture and insertion process, and reporting gaps. The next section discusses the technical problems encountered during the demonstration that led to the loss in data.

3.3.3.8 GNOSC

The AIDE system at the GNOSC was configured to run purely in a global role for the duration of the demonstration. Its mission was to perform demonstration wide correlation and situation awareness. No sensors were actively feeding the machine, and its network was not attacked directly by the white team. All of the information fed into the system came from reports forwarded by the 4 regional systems, and was dependant on both the regional status and local settings. During the majority of the demonstration, at least one regional was reporting to the GNOSC, but frequently several were down at once. Events occurring both at the regional systems and at their local sites were visible at the GNOSC, providing a view of the overall demo attack patterns. This was limited, however, by the amount of downtime at the various regional systems. Correlation also proved problematic, as it required at least 2 regional sites to report a similar event within a relatively short time period, and often an regional would bog down or crash under the demo load, delaying the data until it fell outside of the correlation window. The GNOSC was successful in demonstrating correlation on the last day of the exercise by orchestrating a simultaneous attack of several regional sites that had been reconfigured for better performance.

The GNOSC AIDE performed as expected, with little or no downtime during the demo. The Team performed on the fly upgrades to the box in the middle of the demonstration as part of the overall AIDE performance enhancements, and had no trouble. The SKIP communications from the other systems likewise proved to be exceptionally reliable, and good prior coordination greatly reduced the problems with interactions between SKIP and various firewalls and screening routers.

3.3.3.9 Data Un-captured

The number of 6510 reports prepared, sent, and received was not captured in the database. There is little information in White Cell reporting on 6510 transmissions. This metric will definitely be managed better in the Year Three demonstration.

3.3.4 Year Two Demonstration Technical Problems Encountered

The Year Two demonstration can be divided into two distinct parts, week one (16-20 August) and week two (23-27 August). During week one, most sites suffered from configuration, performance, and database problems. A number of these problems were mitigated in week two, making week two considerably more effective in testing AIDE capabilities. Overall the problems can be categorized as problems with NetRadar, performance problems, configuration problems, and database insert problems. Figure 4.3 shows the numbers of sites affected by each during the course of the two weeks. Please note that certain sites experienced multiple problems.

3.3.4.1 NetRadar

The NetRadar intrusion detection system was installed on all AIDE systems to baseline AIDE data collection capabilities and as a stopgap should a particular site not be equipped with an intrusion detection capability. It was not intended to function as the primary intrusion detection

system. Due to problems with gaining access to ASIM, NetRadar became the sole intrusion detection system at ACC, AFIWC, AFRL, Offutt, and SPACECOM. Due to non-standard JIDS configurations, NetRadar became the primary intrusion detection system at ACOM and Scott. During the demonstration NetRadar also became the primary sensor at ASC. The bottom line is that 8 of the 12 Year Two participants relied on NetRadar as their primary intrusion detection system for the demonstration.

Placement of NetRadar varied from site to site. NetRadar resides on the AIDE box. For example, at Offutt and STRATCOM, NetRadar was located behind the firewall and could only gather internal network data. In both instances NetRadar could not detect intrusion data generated by the Red Team.

This reliance on NetRadar became problematic when, during the demonstration, it was discovered that its OCI BRIDGE was unstable and unreliable. The OCI BRIDGE would go down without warning and without alerting the users to its demise. Though alerts were displayed on the AIDE browser and correlation was performed, with the BRIDGE down, NetRadar intrusion data could not be stored in the Oracle database. For purposes of the post demonstration analysis the AIDE had only partial data to compare against the scripted attacks.

3.3.4.2 Performance Problems

While the instability of the OCI BRIDGE posed the most pervasive problem, the most significant problem in operating AIDE was with its performance. Throughout the demonstration the G-2 expert system locked up due to an inability to process the volume of data produced by certain sensors. This problem was significant at locations running both the Sidewinder firewall and NetRadar. Sidewinder in combination with NetRadar would overload G-2. At LIWA, the site with the greatest number and variety of sensors, G-2 became unusable during high traffic times. Because of the variety of sensors at LIWA, it could not be established whether a single sensor was causing G-2 to lock-up, or whether it was the combination of sensors. The AIDE team triaged the problem at LIWA as being the result of strain on the Sun Sparc Ultra 10's processor created by running NetRadar, receiving a large amount of sensor data and G-2's keeping correlation objects alive over a period of time.

Performance problems were experienced at every AIDE site. AIDE needed to be rebooted multiple times. The reboot process was slow, and data was lost. The performance problem caused data to be deleted from G2 before it could be inserted into the database and before it could be displayed on the browser.

3.3.4.3 Configuration Problems

The location of the AIDE box in relation to the sensors and firewalls and routers proved to present problems during the demonstration. During the first week, routers and firewalls at AFRL, LIWA, and FIWC blocked all attack traffic into the network and prevented AIDE from capturing data. At ASC, AIDE was located behind a switch, which also prevented AIDE from collecting data.

At AFIWC, the AIDE box also served as the Red Team's target. SKIP prevented all connections to the target, hence preventing any attacks against the target. While SKIP worked, attack data could not be collected. Once the problem was correctly diagnosed, AFIWC box reported 73% of all attacks.

One configuration problem beyond the AIDE team's control was that SPACECOM changed their network's domain name the day before the demonstration. This required the reconfiguration of SKIP and elements within the Oracle database and web server resulting in a loss of two days of data.

3.3.4.4 Database Inserts

During the first week of the demonstration, the AIDE team recognized a problem with the Oracle database. Information from G-2 was not reliably inserted into the database. This problem was due to unknown data formats, which caused inserts to fail. Additionally, ACOM's process to insert attack data captured by JIDS into a database did not permit logs to be made available for AIDE to read. Data collected on the USACOM JIDS could not be reported by the AIDE system. The ACOM JIDS flushed data prior to it being sent to AIDE.

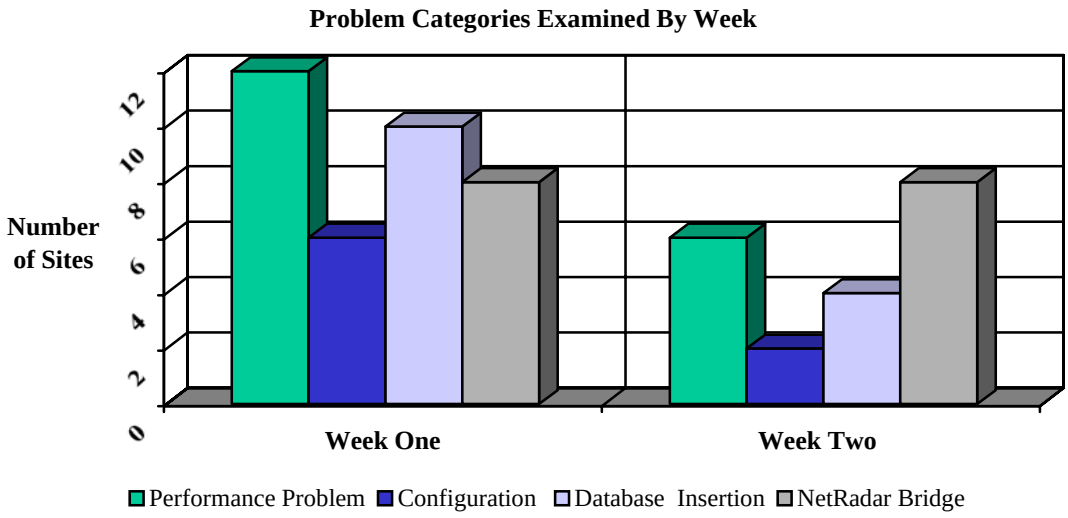


Figure 3-3: Year Two Demonstration Problem Categories Described by Week

3.4 Year Two Feedback

This section is a compilation of user feedback. This feedback was collected in three phases: prior to the demonstration as part of the training program; during the demonstration as part of the White Cell daily reports; and after the demonstration as part of the Year Two Hot Wash.

3.4.1 Pre-Demo (Training) Feedback

The 41 attendees at the AIDE training session were impressed by what the system had to offer; in addition, they shared many constructive comments and suggestions as to how the system can be improved make it an even more useful tool. The "hot topics" that seemed to surface throughout training included the following:

- User interface is awkward and needs some changes
- Reporting process should be streamlined to make it more efficient for the user
- A concern over system performance during periods of heavy traffic
- Desire for inclusion of source/destination IP information and port service information
- Concern with writing local rules

3.4.2 Demo (Site) Comments

The major comment during the Hot Wash was that the concept of integrating the various commercial and government sensors was a first and proved valuable. All users generally agreed that the ability to store the data and query the database was beneficial to their mission of analyzing and reporting on intrusions. While most agreed the concept of the automated 6510 reporting represented a potential a time saver, all agreed that the implementation needed further refinement. Finally, all users agreed that performance problems needed the greatest attention for Year Three.

3.4.2.1 Demo Prep and Execution

All sites were pleased with the test procedures that were implemented as a result of the Test Plan Working group meetings. They believed, in contrast to Year One that the AIDE training was useful. They would, however, like to have a functional AIDE box and training well in advance of the demonstration. Users also expressed a desire to have integration and configuration of supporting systems accomplished earlier in the process rather than waiting until the week before the demonstration. During the execution phase of the demonstration all users saw great value in having the Command White Cell and Red Team co-located. Special kudos went to 2nd Lt Reder, who worked as the liaison between both units. Overall, the sites and the users appreciated the hard work that went into the preparation and execution. They appreciated the predictability of the daily procedures and the ability to call the White Cell and the Red Team with site-specific requests.

3.4.2.2 Three Best

At the end of the demonstration each site was asked to come up with the three best aspects to the AIDE system. The consensus was:

- Viewing multiple sensors in one place and the integration of commercial sensors in particular,

- Having web server access to the database, and
- Automatic 6510 reporting.

3.4.2.3 Three Worst Things

At the end of the demonstration each site was asked to come up with the worst aspects to the AIDE system. The consensus was:

- User Interface difficult to use,
- Performance Problems, and
- Need better data displayed.

3.4.3 Post-Demo (Developer) Comments

This is the feedback from the AIDE developers upon their return from the demonstration.

3.4.3.1 Performance

The problems of performance issues require immediate attention. AIDE needs to be more operationally robust. Hardware capabilities, the Sparc Ultra 10, are a contributor to the overall performance problem. AIDE performance needed to be enhanced to insure better reliability (considerably less lock outs and crashes). System performance needs to be improved. This could be accomplished by better filtering and improved pre-processing of data.

3.4.3.1.1 Sensor Interfaces

The performance problems directly relate to the sensor interfaces. There was a need for a standard configuration of sensors for BRIDGES to work effectively. More advanced on-site testing is needed to ensure the interfaces operated at multi-sensor and single sensor environments behave predictably. To this end, advance notice from sites when they are going to change or upgrade sensors is also needed. Finally, for sensor interface to operate flawlessly, in a security conscious environment encryption for data transmitting from sensor to AIDE box is also needed.

3.4.3.1.2 Communications

Users had a hard time tracking the status of remote connections. To this end, users need better status information and have the ability to automatically reconnect. There was value in the selectable auto or manual reporting capability.

3.4.3.1.3 User Interface

The AIDE user interface performed well. The data display needs to be more meaningful to decrease attack response times. The remote data access performed very well. This capability enabled regional and global level to better understand 6510 reported events. Overall, there is a need to improve the user interface to make it more “User Friendly”. This need was also observed for the method of 6510 reporting.

3.5 Recommendations for Year Three

This section summarizes the AIDE development team's technical response to the issues raised by users. This section concludes with a chart (figure 7.2) derived by D-25 assigning actions and issues for the Year Three development, fielding, and demonstration.

3.5.1 Development Team

The following is an outline of the approach the AIDE Development Team is using to mitigate the problems experienced during the Year Two demonstration. This outline was briefed to the Hot Wash attendees.

3.5.1.1 Performance

To improve AIDE performance the design team is studying the following:

- Redesign architecture and data flow with the expert system,
- Implement a new tap and BRIDGE design, and
- Filtering at BRIDGE and GUI levels.

3.5.1.2 Sensor Interface

To improve the sensor interface the team is currently:

- Working more closely with vendors to improve reliability and data coming into AIDE,
- Making sure new sensors are incorporated earlier in the development process, and
- Redesigning AIDE-to-AIDE communications and the automatic reconnect capability.

The team is also looking at the additional sensors available at the current sites and is planning integrate these sensors for the Year Three demonstration. The additional sensors are displayed in Table 3-7. This figure is based on site surveys conducted in the spring of 1999. Current sites will be contacted individually to verify versions and to add sensors to this list if necessary.

Sensor / Site	NetRadar	JIDS	NetRanger	Real Secure	DTF	Raptor	Sidewinder	TCPWrappers	Sunscreen	ASIM	Gauntlet	CISCO Router
GNOSC	✔	✔						✔				
LIWA	✔	✔	✔	✔	✔		✔	✔				
SCOTT RCERT	✔			✔	✔			✔				
AFIWC	✔							✔				
FIWC	✔		✔	✔	✔			✔				
ASC	✔				✔			✔				
STRATCOM	✔	✔			✔	✔		✔				
ACOM	✔				✔			✔				
Offut AFB	✔				✔		✔	✔				
ACC	✔			✔	✔		✔	✔				
AFRL	✔			✔	✔			✔	✔			
SPACECOM	✔				✔			✔				

Table 3-7: Sensor Matrix Highlighting Additional Available Sensors at Year Two Sites

3.5.1.3 User Interface

The following improvements are planned for Year Three development relative to the user interface:

- User interface taken out of G2,
- A custom Java™ front end will be developed, and
- The User Interface will be tested by users prior to deployment.

3.5.2 Hot Wash Suggestions

According to the D-25 summation of the Hot Wash, the Year Three emphasis of will include issues addressed in Table 3-8. The AIDE analysis team assigned "solution teams" to respond to the individual sites.

Solution Team / Issue Identification	Technical Solution	Management Solution	Training Solution	Implementation Solution
User jury to gain operational perspective	✔			
Revise the management plan to more clearly focus on key deliverables and milestones		✔		
Hold mini-demos throughout the year to improve the system and ensure the sites provide ongoing feedback	✔	✔		✔
Use a spiral development approach for upgrades to the system	✔			
Focus on the areas of correlation, visualization and data reduction	✔			
Consider a minimum standard set of configurations for the sensors but allow some site-specific flexibility	✔			✔
Revisit the site participants for FY00		✔		
Stabilize the existing AIDE box before deploying to any new sites	✔			✔
Need to review the AIDE CONOPS		✔		
Consider possible reengineering of the AIDE box	✔			
Need to develop baseline data on the sensors with respect to the AIDE box performance and measures of success	✔			
Emphasize the deployed JTFs as a customer of the AIDE		✔		

Table 3-8: Matrix of Hot Wash Suggestions and Design Improvements

4 Year Three

Year three would begin on the heels of what was not as successful as expected demonstration at the end of year two in August of 1999. Due to changes in management, changes in software, and the concept of mini demonstrations leading up to a final demonstration year three would ultimately span approximately 18 months until the final demonstration in March of 2001. This period would see significant changes in the AIDE with the complete abandonment of G-2 and considerable portions of the software rewritten from scratch. To further complicate the period some of the initial goals and constraints had been altered including new sites, sensor changes, and more detailed requirements.

4.1 Year Three Development

The primary performance issues with G-2 had been determined prior to the start of the Year Two demonstration and were only confirmed during the course of the demonstration. During the course of Year Three AIDE would change from being described as composed of three parts: the primary interface and operational software written in G-2; the "Bridges" to the sensors, and the Oracle database. By the end of the 18 months that made up Year Three it would be end up being described as "Oracle centric" with the Oracle database being the primary component. Supporting components included correlation, bridges and taps, and the user interface, which would ultimately be developed from scratch in Java. In fact correlation would be performed in G-2 at the outset of Year Three, migrated into C, and ultimately be performed in Java by the time of the final demonstration.

The immediately following section outlines overall goal as a function of the recommendations and lessons-learned from Year One. The remaining subsections detail development in each of the primary parts of AIDE (User Interface, Oracle Database, Bridges; and Correlation) as well as a section devoted to other components relating to AIDE.

4.1.1 Developmental Goals for Year Three

Year Three's system was a significant departure from the system used for the Year Two demonstration.

	Year Three Goal	Lesson Learned	Innovation
Performance	✔		
Normalization and Correlation	✔		
Java User Interface			✔
Hierarchical Reporting			✔
Encrypt		✔	
Visualization		✔	✔
Bridge Development		✔	✔
		✔	
		✔	

Table 4-1: Year Three Goals

4.1.2 User Interface Development

At the end of Year Two the GUI still consisted of primarily a G2 interface with some exploration into 3D rendering in Java. Over the 18 months that would eventually make up the span of Year Three development activity and entirely new graphical user interface would be developed.

4.1.2.1 New Java GUI

An entirely new Java user interface was developed using the basic scheme for display of event data as the G2 interface. Multiple iterations with continued improvement from one version to the next were created. Central to the GUI was the capability to manage the data presented to the user. This consisted of a detailed ability to filter the presented data on a variety of parameters. Later it evolved into separate windows for scrolling data display and “hold-for-analysis”. A variety of capabilities including: Default fields zeroed, additional filtering capability, status bar (times) information, capability to start and monitor bridges from GUI, consistency improvements throughout the menu system, rollup capability, drilldown capability to examine the detail around events, a completely revised incident menu/display/form and capability, as well an entirely new correlator interface was developed, tested, and implemented.

4.1.2.2 Web GUI

The Web GUI became a component in viewing event data as well as allowing the use to configure the AIDE system via a web-based browser. Implemented as part of the Oracle Web Server and built from Oracle tools this capability has been constantly expanded to keep up with the changing database tables and schemas.

4.1.3 Oracle Development

During Year Three the Oracle Database would become the center and key component of the AIDE System with the eventual entire departure of G2. As new data types and capabilities were integrated into AIDE, additional tables and fields were added to support the capability. Many of the functions of AIDE are the direct result of database capability embedded in triggers and SQL code.

4.1.3.1 Oracle Capabilities

Two major activities were implemented in Oracle: hierarchical event queuing, and incident reporting and queuing. The complexities in these activities represent the entire network connectivity capabilities in AIDE.

4.1.3.2 New Tables and Fields

The scope of all the table and field changes over the course of Year Three are too numerous to list here but briefly the following capabilities were supported with extensive table additions or modifications (in addition to the new and modified tables/fields for the Oracle capabilities above): capability for support of multiples of the same sensor type, capability for support of host based event data, and support for all the correlation capability.

4.1.4 Bridge Development

Primarily, all of the Perl Code was done during Year Three development cycle. Numerous iterations of the Real Secure Interface as well as the CMDS/KSE Perl Interface were developed during this time.

4.1.4.1 Updated Bridges

The Real Secure Interface was tested for versions 3.2, 5.0 and 5.5 as well as the Nokia platform. The Raptor TAP was modified due to STRATCOM's issue with it exiting when the log file took longer than normal to rotate out.

4.1.4.2 New Bridges

Snort tap (parser based, signal handling, daemon mode, multiple Snort version support) this included a port to Linux as well as a port to Solaris 7 x86.

4.1.4.3 Encryption

An SSH tunnel capability for encrypting sensors was developed. A complete RSA encryption was implemented in the Bridge/Tap communication.

4.1.5 Correlation Development

Over the course of Year Three significant changes in the correlation system took place. At the outset of Year Three there was still a utilization of G2 to accomplish correlation. G2 not only proved to still be limited in capability for this limited role in AIDE, but also now proved too costly to justify its inclusion in the AIDE system. For these reasons the correlation was redeveloped in C interfacing directly to the Oracle database as data was provided from the Bridges. The C code would examine the new data, compare and make correlations based on the existing data known to the correlation engine, and generate correlated "Event" records to be inserted into the Oracle database for display via the new Java GUI. This correlation engine was totally rewritten in Java prior to the Year three demonstrations and will be the correlation system described in the following subsection.

4.1.5.1 Correlation System

Designed and developed correlation system.

4.1.5.2 Correlation User Interface

4.1.6 Other Aide Development

In addition to the code developed for the AIDE system itself, there was significant additional work done on supporting the AIDE system. Two major areas included the preparation and maintenance of documentation and training material used to familiarize operators with the AIDE system and the construction of software to support the building of an AIDE System.

4.2 Year Three Deployment

To insure that the Year Three demonstration would be successful an effort was made to establish each operational site as soon as possible. The initial plan called for site evaluations to occur prior to April 2000 and installations to occur during the periods from March 2000 through August 2000. Sites, which were evaluated early in the period, and/or previously participating sites would occur earlier in the schedule. In parallel with this would be two Mini-demonstrations to assure that as sites were installed we could successfully accomplish the Final demonstration. These mini-demonstrations were scheduled for April 2000 and August 2000. The final demonstration was to occur in January of 2001.

There were significant issues with the site surveys that delayed the installations. Additionally, the requests for development changes as well as bug corrections continued significantly into the final months of 2000. This meant that the deployment schedule continued past the planned date for a code freeze until sometime in early 2001.

4.2.1 Site Participation

Year Three's site selection consisted of 9 sites. Of these, 6 were from Year Two's initial selection of sites with 6 new sites (of the 6 Year Two sites, 4 were also from Year one). As in

Year Two, the reporting hierarchy consisted of three levels. The local level had AFRL and Dover reporting to JPO as well as STRATCOM, TRANSCOM, and JFCOM reporting to Scott R-CERT. The regional level included Scott R-CERT, JPO, and LIWA all reporting to GNOSC. This is shown in Figure 4-1.

It should be noted that the number of sites fell from twelve to nine (refer to figure 2.1) however AIDE had been implemented and exercised at 4 additional sites (PACOM, QUANTICO, ACC, and JBC) that were not participating in the demonstrations. As in Year Two, we found the personnel at the individual sites knowledgeable on the operation and security of their networks however, it was noted that there was a significant change in personnel with almost no one having experienced the Year One demonstration. We also found that these sites now had a varied understanding of intrusion detection technology. Bounded by cost and service-specific standardization efforts, individual sites continued to deploy a limited number of varied intrusion detection tools.

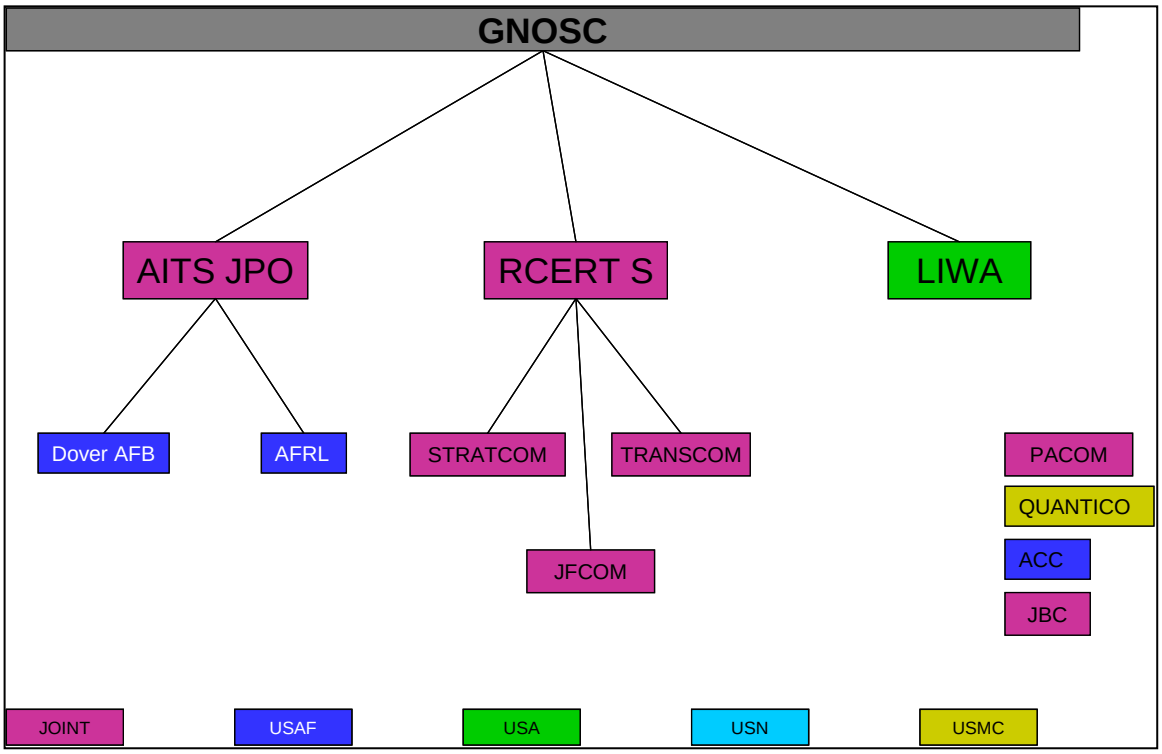


Figure 4-1: Year Three Participants

4.2.2 Sensor Distribution

Year Two brought an additional 4 sensors into AIDE and left only 4 of the initial 10. Table 4-2 shows the sensor distribution for Year Three by site. The new sensors are shaded as well as the new sites.

Site \ Sensor	SNORT	JIDS	ITA	Real Secure	Cisco Router	Raptor	Sidewinder	Emerald	AIDE Correlator
GNOSC	✔								✔
LIWA		✔	✔	✔			✔		✔
SCOTT RCERT		✔		✔		✔			✔
JPO				✔					✔
TRANSCOM	✔	✔							✔
JFCOM		✔							✔
STRATCOM		✔		✔		✔			✔
DOVER	✔						✔		✔
AFRL				✔	✔			✔	✔

Table 4-2: Year Three Sensor Distribution

4.2.3 Site Survey, Installation, and Testing

Installations were hampered by the late arrival of Site Survey Data as well as the changing software.

4.3 Year Three Demonstration

As part of additional measure to assure success of the demonstration, the demonstration schedule was not limited to a single final demo, but included a series of two mini demonstrations.

4.3.1 Demonstration Planning

As stated in the previous section, the initial planning for year three would not rely on a single demonstration, but would consist of two Mini-demonstrations to assure that as sites were installed we could successfully accomplish the Final demonstration. These mini-demonstrations were scheduled for April 2000 and August 2000. The final demonstration was to occur in January of 2001.

4.3.2 Demonstration Execution

In actuality, the execution of demonstrations went as follows:

- Mini-Demonstration 1 – 25-26 April 2000
- Mini-Demonstration 2 – September 7-8 2000
- Mini-Demonstration 3 – February 13-15 2001
- Final Demonstration – March 27-28 2001

Unfortunately, the desire to get sites operational at the earliest date possible through the execution of the mini-demonstrations did not succeed. Sites still had limited resources to expend on the demonstrations and asking them to cooperate in more than one may have mitigated the "last minute" character of the Year Two demonstration by just having them to stretch resources and spend even less time on each of the demonstrations.

The first Mini-Demonstration consisted of 3 sites from Year One: GNOSC, STRATCOM, and AFRL (the only other site to span the entire three years of the ACTD was ACOM which was reorganized and renamed JFCOM). . With limited participation and limited attacking capability this demonstration produced no significant results.

The second Mini-Demonstration was to consist of 7 sites: GNOSC, AFRL, Scott R-CERT, JFCOM, JPO, TRANSCOM and Dover. Neither TRANSCOM nor Dover ended up participating. No additional information was gained from GNOSC or AFRL. As with the first mini-demonstration, the ability to generate attacks was limited. The results of this mini-demonstration we comparable to the first as a result of continued inability to actually generate alerts from the sensors.

The third Mini-Demonstration was to consist of all players in the final demonstration. To offset the continued inability to trigger sensors AFRL had prepared a capability to playback data locally at a site in the purview of the sensors that would assure that the sensor would collect data to feed the AIDE System. Such a system was not used due to the complexity, additional hardware requirements, extra coordination, and added learning curve that needed to take place at each site. As such only half of the attacks succeeded.

The Final demonstration sorted out the issues related to running the local playback capability. This data generation mechanism was relied on totally for data generation. The results of that effort are detailed in the following section.

4.3.3 Demonstration Data

This section contains the technical data extracted from the site databases after the demonstration. It discusses the successes the challenges experienced during the demonstration.

4.3.3.1 Demonstration Overview

AIDE users were able to access and use information relating to 72% of the events generated during the demonstration. This compares favorably with the results of the Year Two demonstration of 15%. Further, this number was significantly affected by the limitations of the sensors. The sensors only captured 73% of the events generated during the demonstration. **Of those events reported by sensors, AIDE captured 99% of the events generated during the demonstration**. AIDE exceeded its Year Three goals. AIDE was able to display attack data in a three-tiered hierarchy. Intrusion data was normalized and intrusion events were correlated at the local, regional, and global levels.

AIDE was able to greatly exceed traditional detection rates for single sensors. Issues remained for AIDE as some sites experienced performance problems. Data display was significantly improved but continues to benefit from refinement. The implemented correlation performed well but additional correlation capability could be added.

4.3.3.2 Demonstration Data

The mini-demonstrations as well as the goal of having the systems up and operational prior to the start of the demonstration avoided the problems encountered in Year Two where the first three days of the demonstration were primarily devoted to repairing technical problems, communications problems, and working out the event generation process. The demonstration lasted two days and all sites remained operational for both days. One site experienced sensor issues. This caused no local events to be reported. Analyzing the demonstration data has been easier than Year Two. The selected analysis data in the data sets from individual site databases were intentionally incomplete to speed up analysis. The fact that the White Cell was running all internal attacks greatly improved their success and detection by the sensors.

The following charts present the demonstration data for the entire 2-day demonstration. The data is presented from three points of view, the number of attacks by site, the number of attacks captured within the database, and the number and types of attacks displayed to the AIDE user. This data is fused to produce a comprehensive view of information available to the user during the demonstration.

4.3.3.3 Attack Generation

Table 4-3 shows the number of attacks by site that were generated against networks monitored by AIDE-connected sensors. This number does not include the scripted attacks planned by the JC2WC. The JC2WC was not present during the demonstration. The sites that did perform the White Team insider attacks documented which attacks they ran and when they were run. Table 4-4 shows the detailed number of total events generated from these attacks by site as a result of those attacks and captured by the sensors.

	27-Mar		
	Ran	Sensor	AIDE
GNOSC			
SCOTT	8	0	0
TRANSCOM	8	8	8
STRATCOM	8	7	7
JFCOM	8	3	3
LIWA	8	8	7
JPO	8	8	8
DOVER	8	8	8
AFRL	8	7	7
Totals	64	49	48

	28-Mar		
	Ran	Sensor	AIDE
GNOSC			
SCOTT	8	0	0
TRANSCOM	8	8	8
STRATCOM	8	8	8
JFCOM	8	2	2
LIWA	8	7	7
JPO	8	8	8
DOVER	8	7	7
AFRL	8	6	6
Totals	64	46	46

Table 4-3: Executed Attack Scenarios by Site showing sensor and AIDE detection

WHITE Team IP's

199.57.6.9

199.57.6.10

RED Team IP's (Attacks were not generated)

199.57.6.11

199.57.6.12

199.57.6.13

Reporting Site	27-Mar-01		28-Mar-01		Total
	WHITE	RED	WHITE	RED	
GNOSC	4182	0	868	0	5050
SCOTT	208	0	206	0	414
TRANSCOM	228	0	242	0	470
STRATCOM	217	0	400	0	617
JFCOM	50	0	48	0	98
LIWA	231	0	391	0	622
JPO	4275	0	812	0	5087
DOVER	4062	0	107	0	4169
AFRL	71	0	128	0	199
Totals	13524	0	3202	0	16726

Table 4-4: Number of Sensor Events by Site

4.3.3.4 Data Capture: Sensors

Figure 4-2 depicts all the attack information collected in the AIDE database that could be mapped to a White Team attack. This data was saved in the Oracle database. Unfortunately in both at Scott R-CERT the intrusion detection system was could not be made operational prior to the start of the demonstration despite the best efforts of all parties involved. In the case of JFCOM the only sensor used was JIDS and this severely limited the amount of data collected.

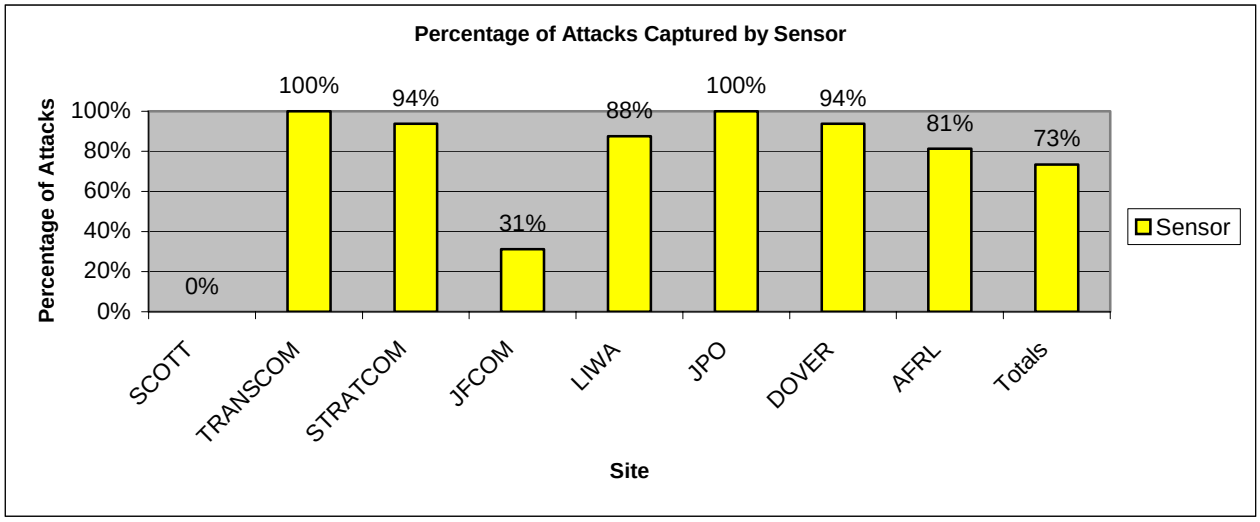
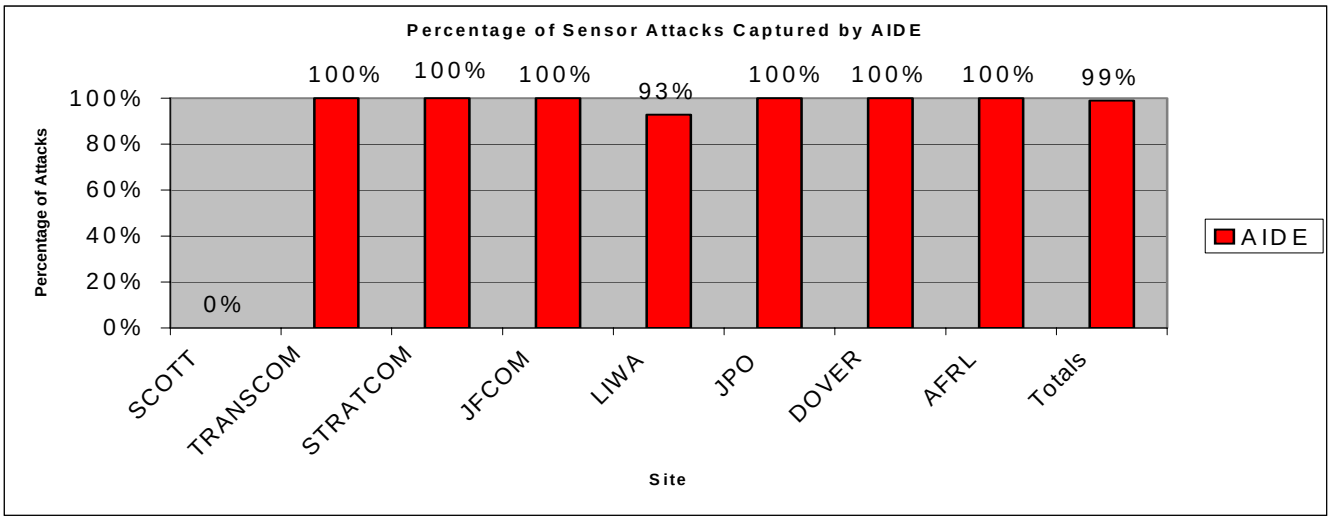


Figure 4-2: Type of attacks captured by the AIDE database broken down by site

4.3.3.5 Data Capture: AIDE Browser

Figure 4-3 depicts event data captured by AIDE and displayed on the AIDE browser. Having validated that the sensors has reported the attack in the previous section and knowing in better detail the actions of the White Team, discerning which data was generated in direct relation to an attack proved substantially easier than any previous demonstration. Problems with incomplete databases and non-descriptive sensor output (which had made previous post demonstration analysis difficult) were not present in this analysis.

Event data is AIDE correlated data, and other sensor data that has been analyzed to match a pattern (in the case of Real Secure, and JIDS), or has been pre-analyzed by the sensor to be considered an attack. This data is saved in the Oracle database as "events". **In comparison to the chart above (Figure 4-2), approximately 99% of the events captured by the sensors were captured by AIDE as an event**. Only a few attacks generated at LIWA could not be fully



correlated with the AIDE event data. In fact the percentage was lower than it should have been as adversely affected by the absolute lack of sensor data from Scott R-CERT.

Figure 4-3: Number of Attacks on the AIDE browser broken down by Site

4.3.3.6 Correlation

The AIDE system at the GNOSC was configured to run in a global role for the duration of the demonstration. The system at JPO was configured to operate as a regional for the demonstration and reported to the GNOSC. Dover was configured to operate as a local site and reported to the JPO.

The following table shows the reduction of data from the correlation at example hosts at the various levels.

Site	Day	Correlated Events were generated for	Raw Sensor Events	Reduced Correlated Events
GNOSC	1	5 of 8 Attacks	4334 from 6 Sites	16
	2	2 of 8 Attacks	319 from 5 sites	5
JPO	1	6 of 8 Attacks	3283 from 2 sites	8
	2	5 of 8 Attacks	492 from 2 sites	8
Dover	1	5 of 8 Attacks	3683	30
	2	2 of 8 Attacks	16	6

Table 4-5: Examples of reduction of events by correlation

It was unclear as to why not all 8 attacks on each day were correlated. It is not certain if all the attacks were actually scheduled to cause correlation nor whether the attacks occurred at the correct time planned time to cause correlation or even if they occurred at all. Further, the

thresholds used for the correlation had only been speculated and not tested in an operational environment. It is believed that the thresholds may have been inappropriate (duration too short and count too high) although this would require a detailed analysis of the data.

Despite this, **the correlation successfully identified over 50% of the planned correlated attacks and significantly reduced the generated raw events by over a factor of 100**.

4.3.3.7 Data Forwarded

The number of forwarded events was captured in the database. By comparing the databases at the sending and sent-to sites a determination can be made as to the ability of AIDE to successfully forward incidents up the chain of command from the local site to the GNOSC. Although this was a problem during the Year Two demonstration, **AIDE successfully forwarded 99% of the intended to be forwarded events**. The following diagram represents the detail of forwarded events from day one of the demonstration.

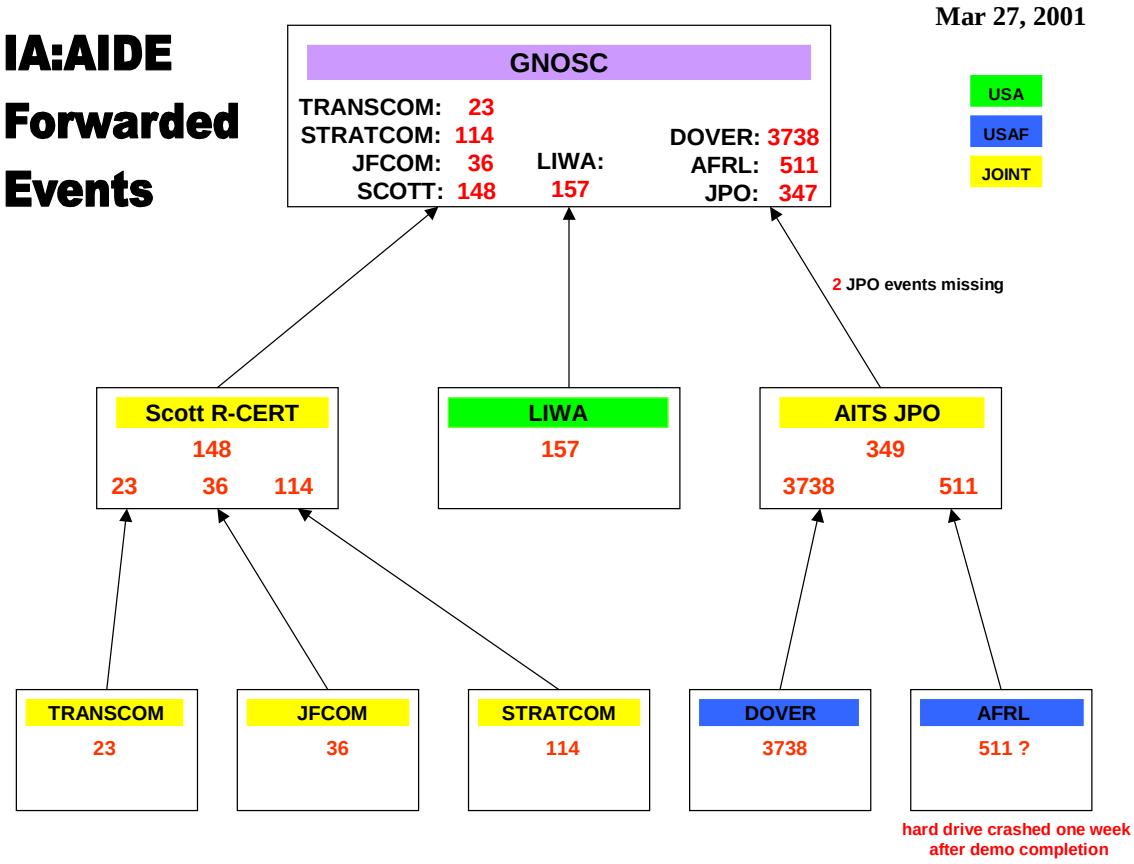


Figure 4-4: Site diagram of forwarded events

4.4 Year Three Feedback

As with Year Two, this section is a compilation of user feedback. This feedback was collected in three phases: prior to the demonstration as part of the training program; during the demonstration as part of the White Cell daily reports; and after the demonstration as part of establishing goals for future work.

4.4.1 Pre-Demo (Training) Feedback

The 17 attendees at the two AIDE training sessions saw a totally new AIDE system; in addition, they shared many constructive comments and suggestions as to how the system could still be improved make it an even more useful tool. The "hot topics" that seemed to surface throughout training included the following:

- Ability to select multiple attributes to filter on rather than one at a time.
- Octet Filtering
- Range Filtering
- Ability for the user to send desired event(s) to another AIDE site.
- Held Events:
    - Count should update
    - Ability to select desired attributes from a given held event & check new events against these attributes. If they match, move the new events to the hold window.
- Incidents listed under a given event should display title & ID.
- WEB Interface source IP drilldown. Destination IP drilldown also needed.
- ORACLE tnsnames.ora file user interface.
- Ability to create an incident from a group of incidents.
- Master clear for all filters which have been added.

4.4.2 Demo (Site) Feedback

A "Hotwash" conference telephone conversation initiated by Ms. Jo Tate of the DISA JPO occurred on April 5, 2001. The sites provided feedback on the demonstration as well as input to future developments desired for AIDE. Some of the comments expressed during that conference as well as in post and pre conference requests for comment are summarized below.

The personnel at the GNOSC are very happy with AIDE. Staff at the (GNOSC) CERT are planning on making and AIDE constellation with snort sensors their real time analysis and reporting capability, and are going to be deploying their own AIDE systems to their regional CERTS. Additionally, staff at GNOSC have been requested by Col. Huffman to train some of the folks on the watch floor in AIDE, and to keep the ACTD AIDE up and taking feeds from some of the regional CERTs. GNOSC personnel did note that AIDE is still sometimes very sluggish.

LIWA provided comments in reference to performance at their site in their particular configuration. They noted that if the AIDE console gets overburdened with alarms, it causes the Oracle database to slow down to such an extent that the perl script running on the Real Secure console receives Oracle errors and crashes. It can be restarted, but there should be some sane error checking routine to preclude data loads from being halted because the AIDE console cannot keep up with events. Further, they pointed out that the AIDE console does not respond well when a full complement (in their case 12 sensors) of Real Secure sensors feed the AIDE database. The console became overloaded with event data and crashed, which again stopped the perl script that feeds the database. As a temporary solution, the scripts were divided into two parts, one, which will synch the RS sensors to the RS Console database, and another script, which then separately queries the RS database and populates the AIDE database. This alleviated some of this concern.

STRATCOM noted considerable benefits from AIDE. For STARTCOM, AIDE provided real-time event detection; sensors feed and display data within seconds of receiving it. The AIDE browser and Web interface GUIs are easy to use, all critical data is displayed as events occur. AIDE made real time event occurrences easier to investigate. AIDE operators were able to react to events in an efficient manner (all relevant data is on the screen). AIDE reporting functions (manual and automatic) were extremely fast; RCERT receives incident reports within seconds - automatic reporting of select priorities saved time and effort. AIDE taps and bridges are easily configurable and maintainable; taps and bridges rarely failed. At STRATCOM, AIDE worked extremely well with Real Secure. **STRATCOM summarized their comments with the statement that they felt AIDE works; it met or exceeded nearly all of the original program goals/objectives. AIDE should be treated as a weapon/mission system and be funded, staffed, and have personnel trained as such.** Minimum site configurations for an effective AIDE system should be defined with a firewall and Real Secure as a minimum for sensors. AIDE should be fielded to all CINCs and incorporated into their daily operations.

Although USTRANSCOM had a more limited utilization of the system during the demonstration, they are willing to participate in the AIDE pilot service program, but must know total support requirements and rules of engagements on system improvements and modifications process. USTRANSCOM feels that the AIDE System performed beyond expectations; however, they feel there is still work that must be accomplished before fielding to support real world, daily operations. They recommend an AIDE CONOP be written and staffed through the CINC. USTRANSCOM supports the AIDE program and will make input to tailor AIDE to our information systems security architecture and operational need.

The other sites provided similar feedback during the hot wash.

4.4.3 Post-Demo (Developer) Feedback

As part of preparation for any future work on AIDE, the developers prepared a spreadsheet of the suggested issues remaining in the system used during the Year three demonstration. A condensed version of this spreadsheet is provided below.

Issues	Explanation (View comments for more detail)	Sub-area
3D Enhancements	Finish/polish current scene, add additional scene for locals?	3D GUI
RELIABILITY	taps/bridges, correlators, gui, everything	ALL
Signature Management	Continual effort by qualified analyst(?). Will sites update manually? Will there be a push or pull used? Requirements	analyst
Tap installer	Executable that would install & configure tap.	Bridge/Tap
Time sync across sensors	Ability to show normalized sensor times in the UI (may not be possible because of Real Secure)	bridge/tap, db?
Standard_bridge block commit	Block commit after n seconds not just n records	Bridges
Better Encryption techniques	Will we continue with SKIP? How will we handle remote GUI's connecting to the dB(8.1.7)? What will we use for bridge-tap?	Bridges,dB?
Backups and Archival of data	Data Wharehouse? Spooled text file? Reguirements!!	dB
Oracle 8.1.5 - 8.1.7	Apache web server, SSL for SQL*NET connections.	dB
WEB vs GUI?	All AIDE configuration would be handled in the JAVA UI. This would elimiate the WEB front end entirely	GUI
Enhanced filtering	Octet, Range, AND/OR, Master Clear, Save filter set, regular expressions	GUI
Lost GUI-dB connection notifier	Would show user when connection to the dB was lost by the GUI	GUI
Multiple select filter selection	Ability to select multiple values for filtering, printing, etc...	GUI
Timer for long look-ups	Provide timer/cancel functionality for long drilldown lookups.	GUI
Boundary traffic back in GUI	This would again show boudary traffice that matches SRCIP with an event.	GUI, dB
Hot IP Management	Hot IP list stored & updateable via the UI. Would alarm when match is found.	GUI, db
No new Sensor Data warning	Warning message when a no new data user defined threshold has been met	GUI, dB
Incident Management	Requirements needed!	GUI, db?
Site Status	Is comm up/down?, last event sent, last incident sent	GUI, db?
Faster GUI/dB	Continual effort needed to identify anything that causes a degradation in performance.	GUI,dB?
User Roles	Do we need Admin & regular user accounts? More than 2? Reguirements!	GUI,db?
Report Management	Ability to run canned reports from the UI. (Crystal Reports)	GUI,db?
Dynamic Data Reduction	How can we reduce browser events? User defined? Rollups like now?	GUI,dB?
Selectable EVENT forwarding	Ability to send a selected EVENT to a remote site.	GUI,dB?
Wizard Build		Installation
TNSnames.ora configuration	Automatic configuration for Oracle's SQL*NET configuration script.	Installation,dB

Table 4-6: Year Three Developer Feedback

4.5 Recommendations for future enhancements

4.5.1 Development Team

In general, the developers acknowledge that work is still needed on easier installs and some performance issues. There is also a concern about reliability; right now the box still seems to require more care and feeding than sites would truly be comfortable with. The database backups

need to be automated, taps/bridges should not need to be restarted quite as frequently. Also, user roles are needed, so that their ordinary users can use the system without being able to modify anything significant.

4.5.1.1 Easy Installation

The developers feel that installations can be improved but some issues such as the scope of AIDE deployment as well as the method of deployment need to be decided prior to extensive work in this area. Fundamental questions as to deciding if AIDE is a software application to be installed on site equipment or if it is a dedicated hardware software system that will be built and delivered to sites must be resolved before considerable work is undertaken in this area.

4.5.1.2 Performance

The developers acknowledge that there are remaining performance issues that must be addressed. What is needed is the establishment of a target threshold as to the number of sensor events per time period that AIDE will be required to handle and how AIDE will degrade if that threshold is exceeded. LIWA showed that sensor event rates in excess of 250,000 per hour (a sustained rate in excess of 70 events per second) uncovers performance problems. Once a threshold is established extensive testing should be undertaken.

4.5.1.3 Reliability

AIDE has proven in Year Three to now be a highly reliable system but considerable improvements could be implemented. Most noted by the Sites were tap issues often on the perimeter of control of AIDE. Network reliability effects are an area, which could benefit from continued development. Additionally, the utilization of RAID for hardware reliability (several AIDE systems have had hard disk failures)

4.5.1.4 Operational Utilities

Utilities for database management such as database backup, offload, and archival and logging for enhanced data recovery remain to be implemented. Although not critical to successful demonstration of operational capability, these capabilities in terms of provided utilities (as well as others) must be implemented for AIDE to be successful in actual operation.

4.5.1.5 User Roles

In its current configuration, AIDE operates from one all encompassing user level. For operation, the need to track AIDE's use to specific operators will be required and must be implemented. Along with this, the notion of user roles such as "Operator", "Administrator", and "Data Analyst" must be incorporated into the system to provide a level of data integrity such that users with inappropriate roles cannot compromise the system.

4.5.2 DISA Compiled

The DISA JPO prepared a compiled list of enhancements as developed from a variety of sources. The following table is taken from that list and shows the scope enhancements expected in the upcoming versions of AIDE. This list was taken from the April 18, 2001 version provided by DISA; more up-to-date versions should be available from them.

Issues	
Enhanced filtering	Octet, Range, AND/OR, Master Clear, Save filter set, regular expressions
Hot IP Management	Hot IP list stored & updateable via the UI. Would alarm when match is found./And a "watched" IP - as in JIDS sensors
Data retrieval / Tap restart	If tap needs to be restarted, need the option to go back and retrieve missed data.
Standard_bridge block commit	Block commit after n seconds not just n records
Faster GUI/dB	Continual effort needed to identify anything that causes a degradation in performance.
Flexible Sensor settings	Ability to select/deselect reporting Sensors. Be able to collect all information - but not forward all information from (ie) an internal test sensor.
Timer for long look-ups	Provide timer/cancel functionality for long drilldown lookups.
Incident Management	Requirements needed!/Accounting / Metrics / Report how many of what kind of event was detected
Incident Report - date and time	Need a date and time stamp for incident reports (Zulu)/Time sent/Time received/Time responded to
Multiple select filter selection	Ability to select multiple values for filtering, printing, etc…
No new Sensor Data warning	Warning message when a no new data user defined threshold has been met
Backups and Archival of data	Data Wharehouse? Spooled text file?
Site Status	Is comm up/down?, last event sent, last incident sent/ Remove old reports / track status
Lost GUI-dB connection notifier	Would show user when connection to the dB was lost by the GUI /Database connection lost / warning message / attempt an auto-connect
Selectable EVENT forwarding	Ability to send a selected EVENT to a remote site.
Maintain local site updates	Create ability to maintain updates added by the local sites - instead of "overwriting" when signature tables are updated with software upgrades/Related to signature management
Priority Modification	Modify priorities to reflect Categories in 6510.1B
Audible Alert	Give audible alert, such as bell, to notify user of Priority 1, etc
Incident Report update notification	Provide notification when incident report has been updated by another site
Delete Incident Report	Provide the capability of an incident report to be deleted by a local site / Related to Incident Management

Multiple terminal display	Problem noted. More than one terminal off an AIDE server: data displays are not consistent.
Signature Management	Continual effort by qualified analyst (?). Will sites update manually? Will there be a push or pull used? Requirements needed.
Boundary traffic back in GUI	This would again show boundary traffic that matches SRCIP with an event./Has to do with getting Raptor FW data into AIDE
Report Management	Ability to run canned reports from the UI. (Crystal Reports)
Proper display of IP's	Source and destination IP's sometimes swapped with JIDS. JIDS or AIDE fix?
Better Encryption techniques	Will we continue with SKIP? How will we handle remote GUI's connecting to the dB(8.1.7)? What will we use for bridge-tap?
AIDE System Time	Problem: Noted the AIDE system time at the bottom of the screen halts for minutes at a time and refreshes at indeterminate times.
Dynamic Data Reduction	How can we reduce browser events? User defined? Rollups like now?
Wizard Build	Simplified Installation
TNSnames.ora configuration	Automatic configuration for Oracle's SQL*NET configuration script.
Screen scroll speed	Screen moves too quickly - and events could not be easily captured
Event description file	Nice to have: Event description file that explains an event - and why it may or may not be an attack/intrusion. (I.e. - Real Secure event description file)
Oracle 8.1.5 - 8.1.7	Apache web server, SSL for SQL*NET connections.
WEB vs GUI?	All AIDE configuration would be handled in the JAVA UI. This would eliminate the WEB front end entirely
Redevelop signatures	Develop signatures (I.e. Known Vulnerabilities)
DB Clean Up	
Event highlight	When an event is highlighted, you cannot deselect the event unless you click on another event
User Roles	accounts? More than 2? Requirements!
Tap installer	Executable that would install & configure tap.
Time sync across sensors	Ability to show normalized sensor times in the UI (may not be possible because of Real Secure)
Auto -refresh on Web Interface	
Multiple instance of COE browser across platforms	Nice to have.
Hour glass	Display hour glass when AIDE is "thinking"
Color code priorities on web browser	Web browser displays priority numbers / but not colors
3D Enhancements	Finish/polish current scene, add additional scene for locals?
Increased interfaces	AIDE must interface with JNMS/HPOv or Remedy
Second "view only" web browser	Nice to have.

Share pictue with Network Mgmt	AIDE must be able to share the picture (to/from) the network management side.
Event Reports Icon	NM able to drill down to icon to pull database data on CND events/reports (pull down from AIDE database)
HPOv / AIDE event drill down	Network device/server degradation or outage causes icon on HPOv network map to change, which triggers an accompanying alert to the CND analysis system (AIDE/OeSP etc.) so that the CND analyst can drill down to identify (correlate) events that might have triggered the problem
	Network device/server degradation or outage causes icon on HPOv network map to change, which triggers an accompanying alert to the CND analysis system (AIDE/OeSP etc.) so that the CND analyst can drill down to identify (correlate) events that might have triggered the problem

Table 4-7: DISA CCB list

5 Conclusion

According to the AIDE ACTD management plan, by the end of the Year Three demonstration the AIDE will reduce false positive reporting and to create a tactical warning capability. The second demonstration proved that we were well on the way to achieving that goal. The goals for the Year Three demonstration were successfully met. AIDE allowed local, regional, and global level analysts to receive, view, and analyze intrusion detection data.

At all nine of the nine sites in Year Three demonstration there was a significant increase in local capability. Throughout the Year Three demonstration each site:

- Detected testing activities (intrusion attempts),
- Provided timely warning,
- Transmitted alerts to higher headquarters.

Performance during the Year Three demonstration in these three areas was outstanding.

- **Of those events reported by sensors, AIDE captured 99% of the events generated during the demonstration**.
- **STRATCOM noted that AIDE truly provided Real time event detection - sensors fed and displayed data within seconds of receiving it.**
- **AIDE successfully forwarded 99% of the intended to be forwarded events.**

Higher headquarters, in this case CERTs/CIRT, were able to drill down to the local site's database. Attack correlation provided an additional layer of analysis to all levels and all sites. **During the Year Three demonstration the correlation successfully identified over 50% of the planned correlated attacks and significantly reduced the generated raw events by over a factor of 100.**

Feedback from the participating sites was positive and constructive. **STRATCOM summarized their comments on the Year Three demonstration with the statement that they felt AIDE works; it met or exceeded nearly all of the original program goals/objectives. AIDE should be treated as a weapon/mission system and be funded, staffed, and have personnel trained as such.** We believe there are a number of opportunities to improve the system while providing a value added to the individual sites and to the global attack visualization endeavor. Cooperation with DISA and AFRL programs is essential to continued success and eventual fielding of AIDE technologies.